

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Eva Lužnik Žnidaršič

**Učenje programiranja za otroke s
pomočjo Raspberry Pi**

DIPLOMSKO DELO

UNIVERZITETNI PROGRAM PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: izr. prof. dr. Janez Demšar

Ljubljana 2015

Fakulteta za računalništvo in informatiko podpira javno dostopnost znanstvenih, strokovnih in razvojnih rezultatov. Zato priporoča objavo dela pod katero od licenc, ki omogočajo prosto razširjanje diplomskega dela in/ali možnost nadaljne proste uporabe dela. Ena izmed možnosti je izdaja diplomskega dela pod katero od Creative Commons licenc <http://creativecommons.si>

Morebitno pripadajočo programsko kodo praviloma objavite pod, denimo, licenco *GNU General Public License, različica 3*. Podrobnosti licence so dostopne na spletni strani <http://www.gnu.org/licenses/>.

Besedilo je oblikovano z urejevalnikom besedil L^AT_EX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Za učenje programiranja v nižjih razredih osnovne šole je zelo prikladno okolje Scratch. V njem programiramo s sestavljanjem blokov, kar je otrokom blizu in jih ne demotivira s sintaktičnimi napakami. Okolje istočasno ponuja možnost za sestavljanje zgodb in iger; probleme, ki jih rešujejo, si lahko otroci zastavljajo kar sami.

Slabost okolja je, da ga otroci po določenem času prerastejo. V diplomski nalogi raziščite eno od možnosti za rešitev tega problema: zamislite si nabor nalog za Scratch, ki ga izvajamo na Raspberry Pi, na katerega so priključene različne LED diode, motorji, senzorji in podobno. Naloge naj bodo blizu otrokom, hkrati pa naj jih peljejo prek različnih konceptov iz programiranja.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisana Eva Lužnik Žnidaršič sem avtorica diplomskega dela z naslovom:

Učenje programiranja za otroke s pomočjo Raspberry Pi

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelala samostojno pod mentorstvomizr. prof. dr. Janeza Demšarja,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 4. septembra 2015

Podpis avtorja:

Gali in Marku.

Kazalo

Povzetek

Abstract

1	Uvod	1
2	Raspberry Pi	3
2.1	O Raspberry Piju	3
2.2	Uporaba Raspberry Pija	4
2.3	Sestava Raspberry Pija	5
2.4	Operacijski sistemi	7
2.5	Prva uporaba Raspberry Pija	7
3	Scratch	9
3.1	Programski jezik Scratch	9
3.2	ScratchGPIO	11
4	Učne ure	13
4.1	Uporabljeni elementi	13
4.2	Uvodna ura	15
4.3	Vklop LED diode	18
4.4	Novoletna smreka	24
4.5	Policijski avtomobil	26
4.6	Glasbeni inštrument	31
4.7	Robotova roka	35

KAZALO

4.8 Twister	39
4.9 Budilka	44
5 Sklepne ugotovitve	49
Literatura	51

Seznam uporabljenih kratic

kratica	angleško	slovensko
GPIO	general purpose input output	splošno namenski vhodno izhodni vmesnik
LCD	liquid crystal display	zaslon s tekočimi kristali
RISC	reduced instruction set computing	računalnik s skrčenim naborom ukazov
HDMI	high-definition multimedia interface	visoko-ločljivostni multimedijski priključek
LED	light emitting diode	svetilna dioda
ACT	active	aktiven
PWR	power	napajanje
FDX	full duplex	polni dupleks
LNK	link	povezava
USB	universal serial bus	vsestransko zaporedno vodilo
DVI	digital visual interface	digitalno vizualen vmesnik
OS	operating system	operacijski sistem

Povzetek

Raspberry Pi je lahek in poceni računalnik v velikosti kreditne kartice. Je odličen pripomoček za spodbujanje računalniške radovednosti pri otrocih in učenje programiranja. Naredili smo poučne naloge z Raspberry Pijem, namenjene učenju osnov programiranja za otroke stare med 8 in 13 let. Uporabili smo programski jezik in okolje ScratchGPIO, ki temelji na programskem jeziku in okolju Scratch, ustvarjenemu za učenje programiranja osnovnošolskih otrok. Scratchevo blokovno programiranje odpravlja nepotrebne težave, s katerimi se srečajo programerji začetniki, hkrati pa je interaktivnost in izgled okolja otrokom zelo všeč. Pripravljene aktivnosti razvijajo logično razmišljanje, spoznajo otroke z osnovnimi konstrukti programiranja in spodbujajo kreativnost.

Ključne besede: učenje programiranja, Raspberry Pi, ScratchGPIO.

Abstract

Raspberry Pi is a lightweight and inexpensive computer the size of a credit card. It is an excellent tool for encouraging computer curiosity in children and teaching them to program. We have created educational activities using Raspberry Pi aimed for learning the basics of programming for children aged between 8 and 13. We used the programming language and environment ScratchGPIO that is based on the programming language and environment Scratch created for teaching programming to elementary school children. Scratch uses blocks that eliminate unnecessary difficulties encountered by novice programmers, while the interactivity and appearance of the environment attracts the children. Prepared activities stimulate logical thinking, teach basic programming constructs and encourage creativity.

Keywords: teaching programming, Raspberry Pi, ScratchGPIO.

Poglavje 1

Uvod

Raspberry Pi je računalnik v velikost kreditne kartice v prvi vrsti namenjen otrokom. Komponente strojne opreme niso skrite, kot pri modernih telefonih, prenosnih računalnikih in drugih elektronskih napravah, ampak so vidne in otipljive. Dostopnost komponent pri otrocih spodbudi radovednost in omogoči raziskovanje računalništva. Raspberry Pi je bil narejen z namenom izobraževanja učencev osnovnih in srednjih šol in s tem enostavnejšega in cenejšega spoznavanja osnov računalništva in programiranja. [12]

V okviru diplomske naloge bomo naredili poučne aktivnosti, namenjene staršem, učiteljem, starejšim bratom in sestram, ki bi želeli otroke spoznat z osnovami programiranja. Namen aktivnosti je razvijanje logičnega razmišljanja, učenje osnovnih programskih konstruktov in spodbujanje otroške kreativnosti. Aktivnosti bomo preizkusili z otroki starimi med 8 in 13 let, ki obiskujejo 3., 5. in 7. razred osnovne šole. Otroci med 8. in 13. letom starosti bodo ciljna skupina aktivnosti. Izmisлити si moramo naloge, ki bodo pritegnile pozornost te starostne skupine in jim približale učenje programiranja. Naloge ne bodo zahtevale predznanja, težavnost nalog se bo postopoma stopnjevala.

Pri nalogah se bo uporabljalo programsko okolje ScratchGPIO, ki temelji na proramskem okolju Scratch. Okolje je ustvarjeno prav za učenje programiranja otrok med osmim in šestnajstim letom starosti. Otroci v tej starostni

skupini imajo različno znanje tipkanja, kar pomeni, da imajo lahko nekateri težave pri tipkanju ukazov. Pri uporabi Scratcha do teh težav ne pride, saj so ukazni bloki že ustvarjeni, otroci jih zgolj povlečejo v glavno okno. Scratchevo blokovno programiranje odpravlja tudi težave s sintaktičnimi napakami, ki jih imajo lahko programerji začetniki v tradicionalnih računalniških programih.

Poglavje 2

Raspberry Pi

2.1 O Raspberry Piju

Raspberry Pi je zelo majhen, lahek in poceni računalnik. Z izjemo modela A+ so vsi modeli tako prve kot tudi druge generacije veliki 8.56 x 5.65 cm in tehtajo 45 g. Model A+ je še nekoliko manjši, saj meri 6.5 x 5.65 cm, in tehta zgolj 23 g. Raspberry Pi je zmožljiva majhna naprava, ki ljudem vseh starosti omogoča raziskovanje računalništva in učenje programiranja. Uporabljamo ga lahko za iste stvari kot običajen računalnik, na primer brskanje po internetu, igranje računalniških iger, gledanje video posnetkov in pisanje dokumentov. Zaradi nizke cene ga lahko uporabljamo tudi namesto mikrokontrolerjev, routerjev in podobno. Lahko ga uporabimo za vrsto različnih projektov, od glasbenih naprav in vremenskih postaj, do ptičjih hišic z infra rdečo kamero. Prvoten namen Raspberry Pija je, da se predvsem učenci osnovnih in srednjih šol z njim izobražujejo in s tem enostavneje in ceneje spoznajo osnove računalništva. [9] [8]

Ideja majhnega in poceni računalnika za otroke se je pojavila leta 2006. Eben Upton, Rob Mullins, Jack Jang in Alan Mycroft, zaposleni v računalniškem laboratoriju Univerze v Cambridgeu, so opazili razliko v znanju računalništva in programiranja pri dijakih, vpisanih na njihov program. Razliko so pripisali učnemu načrtu računalništva v obveznem šolanju, ki predvideva zgolj

učenje uporabe programa Word in Excel. Poleg tega se je končalo obdobje Dot-com booma. Tretji problem pa se jim je zdel, da so Amigo, BBC Micro, Spectrum ZS in Commodore 64, na katerih so se ljudje zgodnejših generacij naučili programirati, zamenjale igralne konzole in domači računalniki. Obenem računalniki postajajo vedno dražji, kar oteži eksperimentiranje s programiranjem.

Skupina se je med letoma 2006 in 2008 lotila izdelave poceni mini računalnika in ustvarili so veliko prototipnih verzij današnjega Raspberry Pija. Procesorji za mobilne naprave so postajali vedno bolj cenovno ugodni in zadosti zmogljivi, da so omogočili uporabo multimedije, ki se je skupini zdelo pomembna funkcija, da bo otrokom naprava všeč. S temi procesorji je projekt poceni mini računalnika postal bolj realen. Leta 2008 so ustanovili fundacijo Raspberry Pi in 3 leta kasneje je bil Raspberry Pi model B že v masovni produkciji. V dveh letih so uspeli prodati več kot 2 milijona enot. [14] [4]

2.2 Uporaba Raspberry Pija

Raspberry Pi je možno uporabljati na vrsto različnih načinov. Je dovolj zmogljiv, da ga je možno uporabiti za stvari, ki jih ponavadi počnemo z navadnim računalnikom. Omogoča nam, na primer, brskanje po spletu, predvajanje video posnetkov, ustvarjanje razpredelnic, igranje računalniških iger in pisanje besedil. Poleg tega ima Raspberry Pi možnost interakcije z zunanjim svetom, kar omogoča izdelavo različnih digitalnih ustvarjalnih projektov.

Na operacijski sistemu Raspbian imamo na voljo vrsto programskih oprem, med njimi vizualno programsko orodje Scratch za ustvarjanje animacij in iger, vsestranski programski jezik Python in Sonic Pi, ki omogoča ustvarjanje glasbe s programiranjem. Na Raspbianu se nahaja tudi Linux terminal, ki sprejema in izvršuje ukaze. Na voljo nam je tudi verzija Minecrafta, ki jo je možno programirati s programskim jezikom Python in preko pinov GPIO interaktivirati z zunanjim svetom. Raspbian ima tudi računalniške igre, ki jih je možno igrati, pregledati njihovo kodo, jo spremeniti in narediti lastno

igro. Lotimo se lahko tudi urejanja spletnih strani preko sistema WordPress in reševanja matematičnih problemov s platformo Mathematica. Uporabljamo lahko vse programske jezike, ki delujejo na ARMv6. Na Raspberry Piju so privzeto nameščeni Python, C, C++, Scratch, Ruby in Java. Za začetnike fundacija Raspberry Pi priporoča programski jezik Python in za mlajše otroke Scratch.

Projekti, ki so jih ustvarili uporabniki Raspberry Pija, so zelo raznovrstni, zanimivi in kreativni. Tako je na primer uporabnik grahamgelling na spletnem portalu Instructables objavil podrobna navodila, kako izdelati klubsko mizico z arkadnimi igrami. Uporabil je zaslon LCD, priklopljen na Raspberry Pi preko kabla HDMI, krmilno ročico in gumbe priklopljene na GPIO. Na Raspberry Pi je namestil programsko opremo MAME (Multiple Arcade Machine Emulator), ki omogoča igranje klasičnih arkadnih iger na sodobnejših računalnikih. Skupina računalničarjev z Univerze v Southamptonu je med seboj povezala 64 Raspberry Pijev, tako da so ustvarili cenejšo verzijo superračunalnika. Zmogljivost sicer ni bila primerljiva z današnjimi superračunalniki, vendar pa so demonstrirali, da je možno narediti takšen sistem.[1] [3]

Prav tako je uporabnik willq44 na portal Instructables objavil navodila, kako narediti aplikacijo za hišno avtomatizacijo. Izdelal je spletno aplikacijo, ki nadzira luči v stanovanju. Raspberry Pi je povezan v internetno omrežje in z brezžičnim daljinskim upravljalnikom, ki je preko priključkov GPIO povezan z Raspberry Pijem, ugaša in prižiga luči. [7]

2.3 Sestava Raspberry Pija

Na plošči Raspberry Pija se nahaja procesor, več vhodov in izhodov ter statusne LED diode. Raspberry Pi prve generacije ima 32 bitni enojedrni procesor tehnologije ARM11. Model A ima 256 MB delovnega pomnilnika, Modela B in B+, ki sta nekoliko dražja, imata 512 MB delovnega pomnilnika. Takšen procesor se najde tudi v starejših telefonih kot na primer iPhone 3G

oznaka	barva	pomen
ACT	Zelena	Prižgana, ko dostopamo do kartice SD.
PWR	Rdeča	Predstavlja priključitev na napajanje.
FDX	Zelena	Prižgana, če je omrežni adapter full duplex.
LNK	Zelena	Predstavlja omrežno aktivnost.
100	Rumena	Prižgana, če je omrežna povezava 100 Mbps.

Tabela 2.1: Pomen statusnih LED diod

ali elektronskemu bralniku knjig, Kindle 2. Raspberry Pi ima komponentni vhod / izhod za kompozitni video signal, na katerega lahko s kablom RCA priključimo televizijo ali monitor. V primerjavi s HDMI je kvaliteta tega videa precej nižja. Vsi Raspberry Pi modeli imajo 3.5 mm analogni priklop za zvok, na katerega lahko priklopimo slušalke ali zvočnike. Na plošči se nahaja tudi 5 statusnih LED diod, katerih pomen je opisan v tabeli 2.1 .

Modela A in A+ imata zgolj 1 izhod USB 2.0, model B ima 2, model B+ in Raspberry Pi druge generacije pa 4. Na izhode USB lahko priklopimo na primer tipkovnico, računalniško miško in zunanji pomnilnik. Modela B in B+ ter Raspberry Pi druge generacije imajo vhod ethernet, tako da lahko priključimo kabel ethernet in se povežemo v omrežje. Vsi modeli Raspberry Pija imajo izhod HDMI, ki omogoča video in avdio izhod. Na HDMI lahko s kablom HDMI priklopimo televizijo ali monitor. Ker večina monitorjev nima vhoda HDMI, bomo v večini primerov potrebovali tudi pretvornik VGA ali pretvornik DVI. Raspberry Pi ima vhod micro USB, na katerega priklopimo kabel za napajanje. Uporabljamo lahko tudi kartice SD za shranjevanje podatkov. Modela A in B prve generacije uporabljata običajen SD, A+, B+ in modeli druge generacije pa mikro SD. Raspberry Pi ima priključke GPIO, ki jih uporabljamo za branje gumbov, stikal in upravljanje LED diod, motorjev, relejev in podobno. [1]

2.4 Operacijski sistemi

Obstaja veliko različic operacijskih sistemov za Raspberry Pi. Priporočena je uporaba operacijskih sistemov Linux, vendar obstajajo tudi drugi. Najbolj poznana in priporočljiva različica operacijskega sistema Linux za Raspberry Pi je Raspbian, saj ima največ podpore s strani skupnosti in fundacije Raspberry Pi. Poleg Raspbiana obstajajo tudi druge rešitve; za Raspberry Pi so prilagojeni še na primer RISC OS, OPENELEC, PINET, OSMC in UBUNTU. Operacijski sistem za Raspberry Pi moramo namestiti na kartico SD, saj naprava nima notranjega pomnilnika. Fundacija Raspberry Pi je za pomoč začetnikom ustvarila namestitveni sistem NOOBS (New Out Of the Box Software), ki omogoča enostavno namestitev operacijskega sistema na Raspberry Pi. [8]

2.5 Prva uporaba Raspberry Pija

Ob prvi uporabi moramo najprej na kartico SD namestiti enega izmed operacijskih sistemov prilagojenih za Raspberry Pi. Za začetnike je najbolj priporočljiv namestitveni sistem NOOBS, ki omogoča enostavno namestitev operacijskega sistema. Na njem so namestitvene datoteke za različne Raspberry Piju prilagojene operacijske sisteme. Ob prvi uporabi izberemo, katerega želimo uporabiti in kasneje tega ni več možno spremeniti, ne da bi ponovno namestili celoten sistem.

Ko je operacijski sistem naložen na kartico SD, jo vstavimo v Raspberry Pi. Računalniško miško in tipkovnico priključimo na vhode USB. Raspberry Pi je potrebno priključiti na prikazovalnik, najbolj priporočljivo je uporabiti zaslon, ki ga lahko priklopimo s kablom HDMI. To nam omogoči sliko visoke kvalitete, hkrati pa omogoči, da se poleg video signala prenese tudi avdio. Če naš prikazovalnik nima vhoda HDMI, ima pa DVI, lahko za povezavo uporabimo pretvornik HDMI-DVI, vendar DVI ne omogoča prenosa avdio signala. Video signal lahko prenašamo tudi preko kabla RCA, vendar je v tem primeru kvaliteta slike slabša in ne omogoča prenosa avdio signala.

Zvok lahko prenašamo bodisi preko kabla HDMI bodisi s priklopom naprave za predvajanje avdio signala na 3.5 mm analogni priklop. Če želimo, lahko napravo priključimo na žično omrežje preko ethernet vhoda.

Ko imamo vse potrebno povezano na napravo, lahko napajalnik priklopimo v vtičnico. Raspberry Pi nima gumba za vklop, tako da se ob priklopu napajalnika in pravilni namestitvi operacijskega sistema na zaslonu avtomatsko prikaže slika. Če smo na kartico SD naložili namestitveni sistem NOOBS, bomo ob prvi uporabi izbrali operacijski sistem, ki ga želimo uporabljati. Priporočena je uporaba operacijskega sistema Raspbian. Po izbiri operacijskega sistema se bo le-ta začel nalagati, kar zna trajati kar nekaj časa. Prikazal se bo tudi meni z nastavitvami, kjer lahko nastavimo uro, datum in drugo. Za dostop do namizja, je potrebno vpisati uporabniške podatke. Privzeto uporabniško ime za vpis v Raspbian operacijski sistem je pi z geslom raspberry. Iz varnostnih razlogov ob vpisu gesla na zaslonu ni možno zaslediti tipkanja. Po vpisu uporabniških podatkov lahko vtipkamo startx in naložil se bo grafični uporabniški vmesnik. [2]

Da uporabniških podatkov ni potrebno vedno znova vpisovati, lahko nastavimo avtomatičen vpis in avtomatičen dostop do namizja. To storimo v terminalu z urejanjem datoteke inittab. V terminalu vpišemo “sudo nano /etc/inittab” in odprla se nam bo datoteka inittab, v kateri pred “1:2345:respawn:/sbin/getty 115200 tty1 “ postavimo #. V datoteko napišemo “1:2345:respawn:/bin/login -f pi tty1 </dev/tty1 >/dev/tty1 2>&1”, kar omogoči vpis v uporabnika “pi” brez avtentikacije. Datoteko nato shranimo in zapremo. Zdaj lahko naredimo še avtomatično dostopanje do namizja. V terminal zapišemo “sudo raspi-config”, kar bo odprlo meni z nastavitvami. Izberemo “Enable Boot to Desktop/Scratch”, nato še “Desktop Login as user pi at the Graphical Desktop”, zaključimo in ponovno zaženemo Raspberry Pi.[10]

Poglavje 3

Scratch

3.1 Programski jezik Scratch

Scratch je vizualen programski jezik namenjen učenju programiranja. Ustvarjen je predvsem za otroke med osmim in šestnajstim letom, čeprav ga uporabljajo tudi starejši. V njem lahko ustvarjamo zelo različne projekte, na primer računalniške igre, interaktivne časopise, virtualne izlete, naravoslovne simulacije, interaktivne zgodbe, rojstnodnevne čestitke, računalniške animacije in interaktivne vadnice. Dnevno uporabniki Scratcha na spletni portal scratch.org naložijo več kot 1500 novih projektov s prostodostopno kodo, ki se jo da deliti in spreminjati. Uporabniki se s programiranjem in deljenjem interaktivnih projektov naučijo pomembnih matematičnih in računalniških konceptov. Razvijajo tako sistematično kot tudi kreativno razmišljanje.

Programerski začetniki imajo v tradicionalnih računalniških programskih jezikih težave s sintaktičnimi napakami. Scratchevo blokovno programiranje te težave odpravlja in je zaradi tega veliko bolj prijazno za programerja začetnika. Scratch omogoča ustvarjanje in nadzor grafičnih objektov, imenovanih figure, ki se premikajo po zaslonu. Omogoča tudi integracijo grafike, fotografij in zvočnih datotek za ustvarjanje multimedijskih aplikacij. Scratch je enostaven za učenje. Ponavadi je potrebno 6 učnih ur, da se programer začetnik nauči glavnih programerskih konceptov in začne ustvarjati

lastne igre in multimedijske projekte. Začetnik lahko v Scratchu naredi vrsto različnih aplikacij, na primer preprosto igro, kjer igralec nadzira figuro, ki lovi drugo figuro po ekranu, animacijo, kjer figure predstavljajo črke programerjevega imena in se vsaka med njimi vrti in ob klicu sproži predvajanje zvoka in kviz, ki prikazuje slike in vprašanja z odgovori, ki se odzivajo na uporabnikov klik. Izdelovanje aplikacije je ponavadi sestavljeno iz risanja oziroma izbire privzetih figur, nalaganja oziroma ustvarjanja lastnega ozadja, snemanja zvočnega posnetka, vstavljanja besedila v aplikacijo, premikanje figur po zaslonu z različnimi grafičnimi ukazi in digitalnimi učinki na grafično figuro. S programiranjem v Scratchu se lahko naučimo osnovne principe računalniškega programiranja kot na primer ustvarjanje zaporedja ukazov, uporabo pogojnih stavkov, ponavljanj, premikanja figure po ekranu in odzivanja na okolje, odzivanje na tipkovnico in druge dogodke ter uporabe spremenljivk.

Programsko okolje Scratch je osnovano na istih didaktičnih načelih kot programski jezik LOGO. Leta 1967 so Seymour Papert in kolegi z ene vodilnih ameriških univerz MIT z namenom izobraževanja ustvarili programski jezik Logo, ki je uporabljal takoimenovano želvjo grafiko. Uporabnik je z različnimi ukazi premikal majhnega robotka, imenovanega želva in s tem risal grafične linije na ekran. Logo je bil uporabljen v mnogih šolah v zadnjih treh desetletjih, vendar v nasprotju s pričakovanji ustvarjalcev pri otrokih in učiteljih ni buril domišljije. Skupina Lifelong Kindergarten, ki deluje v medijskem laboratoriju univerze MIT, je nadaljevala z razvojem programskega jezika Logo in nastal je StarLogo, ki uvaja blokovno programiranje. Grafični bloki predstavljajo program in odpravljajo tipkarske in sintaktične napake. Skupina je sodelovala pri vodenju mladinskih računalniških klubov in ugotovila, da so otrokom najljubše multimedijske aplikacije, ki vsebujejo slike in zvok. To je privedlo do nastanka Scratcha. [13]

3.2 ScratchGPIO

ScratchGPIO je posebna verzija programskega okolja Scratch, ki je namenjena nadzoru priključkov GPIO. Z okoljem ScratchGPIO lahko Raspberry Pi uporabimo za nadzor diod LED, motorjev, senzorjev, stikal, itd. To otrokom omogoči lažje programiranje fizičnih projektov, brez uporabe težjih programskih jezikov kot so na primer Processing, Python in C. Programsko okolje ScratchGPIO je ustvaril Simon Walters in je prosto dostopno. [3] Okolje ni med privzetimi, zato ga je potrebno namestiti. Za namestitev je potrebno Raspberry Pi povezati v internetno omrežje s kablom Ethernet. V terminal nato zapišemo ukaz “`wget http://bit.ly/1wxrqdp -O isgh7.sh`”, ki bo zagnal namestitveni program za ScratchGPIO verzije 7. Nato kličemo še ukaz “`sudo bash isgh7.sh`” in programsko okolje ScratchGPIO 7 nam bo na voljo za uporabo. [5]

Poglavje 4

Učne ure

V naslednjem poglavju bomo opisali aktivnosti, namenjene učiteljem, staršem, starejšim sestram, bratom in drugim, ki želijo učiti otroke osnov programiranja. Na začetku so predstavljeni osnovni elementi, ki jih pri učnih urah uporabimo - upor, dioda, LED dioda in povezovalna plošča. Učenje se prične z uvodno uro, na kateri otrokom predstavimo Raspberry Pi. Nato sledijo naloge, s katerimi otroci spoznajo osnovne programske koncepte. Težavnost nalog se postopoma stopnjuje. Začnemo s čisto preprostim prižiganjem LED diode, ki od otrok ne zahteva nikakršnega predznanja. Predvideno je, da se naloge rešuje zaporedno, saj je pri nekaterih priporočeno predznanje, pridobljeno pri prejšnjih nalogah. Posamezna ura je predstavljena z učnimi cilji, potrebščinami in navodili za učitelja, kako naj naloge predstavi in daje pojasnila otrokom. Ponavadi imajo naloge fizični in programerski del.

4.1 Uporabljeni elementi

4.1.1 Upor

Ena najpomembnejših in najbolj uporabljenih elektronskih komponent je upor, ki je ključna komponenta večine elektronskih vezij. Upori imajo električno upornost, ki omejuje tok v vezju. Večja je upornost, manjši je električni tok in obratno. Gre za pasivno komponento, kar pomeni, da energijo

zgolj porablja in je ne proizvaja. V vezje se jih doda kot dopolnilo aktivnim komponentam, kot na primer operacijskim ojačevalnikom, mikrokontrolerjem in drugim integriranim vezjem. Uporablja se jih za omejevanje električnega toka in deljenje napetosti.[6]

4.1.2 Dioda

Ena izmed najbolj uporabljenih polprevodniških komponent je dioda. Glavna naloga diode je nadzor smeri električnega toka. Tok lahko teče skozi diodo zgolj v eno smer. Če je napetost skozi diodo negativna, le ta ne prepušča električnega toka. Dioda ima dva priključka; pozitivni se imenuje anoda, negativni pa katoda. Električni tok lahko teče od anode do katode, v obratno smer pretok ni možen.[6]

4.1.3 LED dioda

Zelo uporaben polprevodniški element za naše projekte je tudi LED dioda. Nahaja se v naših telefonih, avtomobilih in celo domovih. Po izgledu se zelo razlikujejo. Imajo zelo različne oblike, velikosti, barve. So vrsta diode, ki električno energijo pretvori v svetlobo. LED diode so kot majhne žarnice, vendar potrebujejo precej manj energije, da svetijo in se zato tudi manj segrejejo. Gre za vrsto diode, tako da tudi LED diode prepuščajo električni tok zgolj v eno smer. Pozitivna stran, anoda, je označena z daljšim priključkom, negativna, katoda pa s krajšim. Električni tok potuje od anode do katode. LED dioda sveti zgolj takrat, ko skozi njo potuje električni tok.[6]

4.1.4 Povezovalna plošča

Povezovalna plošča je testna oziroma prototipna plošča, ki ima luknje, na spodnji strani povezane z bakrenimi trakovi. V luknje lahko vstavimo različne komponente in jih med seboj povežemo. Uporaba povezovalne plošče nam odpravi potrebo po spajkanju. Najpogosteje so uporabljene za prototipiranje. Nanje lahko priklopimo preprosta vezja kot tudi kompleksnejša. Če

plošča postane premajhna za naš projekt, jo lahko povežemo z drugo ploščo. Ko komponento vstavimo v povezovalno ploščo, bo električno povezana z vsemi drugimi komponentami, ki so vstavljene v isti vrstici. Vrstica ima dva bakrena trakova, ki imata 5 vhodnih lukenj in sta ločena z vdolbino, ki vertikalno ločuje povezovalno ploščo na dva dela. Tako lahko hkrati med seboj povežemo največ 5 komponent. Poleg horizontalnih vrstic imajo povezovalne plošče ponavadi na obeh straneh tudi vertikalne stolpce, ki so povezani z železnimi trakovi. Te so namenjene stalni pozitivni napetosti (+) in ničli (-, GND). [6]

4.2 Uvodna ura

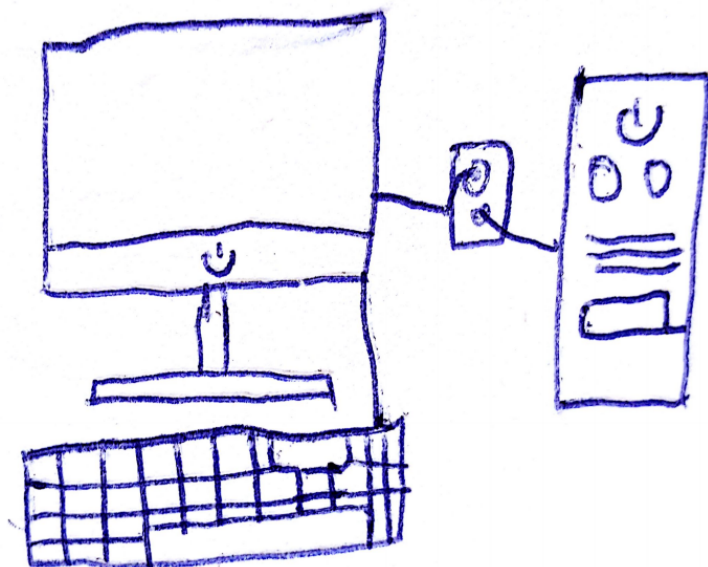
4.2.1 Učni cilji

Namen uvodne ure je, da otrok spozna osnove uporabe Raspberry Pi ja. V njej otrok izve, kaj je računalnik, kaj strojna in kaj programska oprema. Nauči se poimenovati in razumeti namen osnovnih delov računalniške strojne opreme. Nauči se zagnati Raspberry Pi, vpisati uporabniške podatke in dostopati do namizja.

4.2.2 Potrebujemo:

- Raspberry Pi
 - zaslon z vhodom HDMI
 - kabel HDMI
 - kartico SD
 - računalniško miško z izhodom USB
 - računalniško tipkovnico z izhodom USB
 - kabel za napajanje
 - namizni računalnik z opremo (zaslon, miška, tipkovnica,...)

Na kartico SD vnaprej naložimo operacijski sistem Raspbian in poskrbimo za avtomatičen vpis in dostop do namizja, brez vpisa uporabniških podatkov.



Slika 4.1: Otroška risba računalnika.

Postopek je opisan v poglavju Prva uporaba Raspberry Pija.

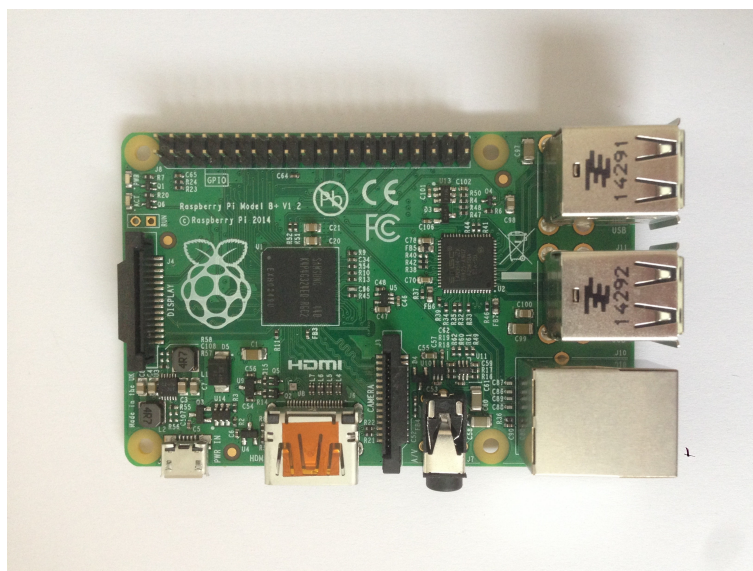
4.2.3 Učna ura

Prosimo enega od otrok, naj na tablo nariše računalnik 4.1.

Nato pokažemo otrokom Raspberry Pi 4.2 in jih vprašamo, ali je tudi to računalnik.

Otrokom pojasnimo, da je Raspberry Pi tudi računalnik. Je sicer veliko manjši, lažji in cenejši od tistih, ki smo jih vajeni, ampak prav tako računalnik.

Otrokom pokažemo vso opremo - Raspberry Pi, zaslon, kabel za priključitev zaslona, računalniško miško, tipkovnico, kartico SD in napajalnik. Zatem jim damo nalogo, da skušajo poimenovati posamezne dele opreme in ugotoviti čemu služijo. Zaslon je namenjen prikazovanju informacij, s kablom HDMI pa ga priklopimo na Raspberry Pi. Kartica SD je namenjena shranjevanju podatkov. Računalniška miška in tipkovnica sta otrokom dobro



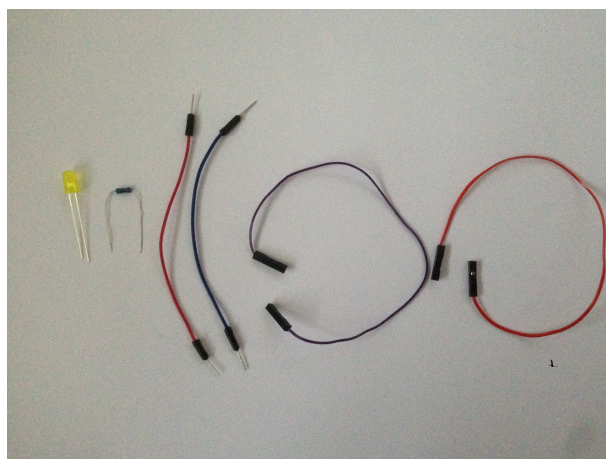
Slika 4.2: Raspberry Pi.

poznana. Otroci bodo z veseljem priklapljali opremo na Raspberry Pi, tako da jih lahko prosimo, da vsak priklopi en del, ga poimenuje in razloži njegov namen. Povemo jim, da so to deli strojne opreme. To so fizični deli, ki jih lahko vidimo in se jih dotaknemo. [8]

Otrokom zdaj pokažemo namizni računalnik in jih prosimo, da poimenujejo dele opreme in razložijo čemu so namenjeni. Otroci pogosto mislijo, da je zaslon najpomembnejši del in da je to računalnik. Razložimo jim, da zaslon zgolj prikazuje informacije, pomembno delo pa opravi računalnik.

Prosimo jih, naj vse potrebno priključijo na Raspberry Pi, pri čemer lahko ponovijo, kar so se naučili. Ob priklopu napajalnika v vtičnico se bo na zaslonu avtomatsko prikazala slika, saj Raspberry Pi nima gumba za vklop kot smo ga vajeni pri osebnih računalnikih. Otrokom razložimo, da kartica SD hrani informacije o tem, kako se mora Raspberry Pi zagnati. Če je ne bi priključili, se ne bi zagnal.

Ker smo poskrbeli za avtomatičen vpis, se bo ob zagonu na zaslonu pojavilo namizje. Otrokom pokažemo nekaj aplikacij, ki jih najdemo na namizju. Te aplikacije so programska oprema, ki se jih za razliko od strojne opreme



Slika 4.3: Pripomočki za vklop LED diode.

ne moremo dotakniti.

4.3 Vklop LED diode

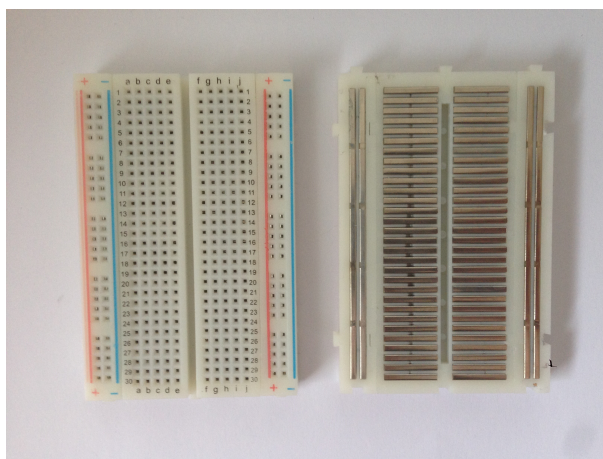
4.3.1 Učni cilji

Tokrat se bodo otroci naučili prižgati LED diodo in jo ugasniti. Prav tako spoznajo pine, GPIO in programsko okolje ScratchGPIO. Naučijo se, kaj je vhod in kaj izhod ter kako povezati elemente, da bo električni krog sklenjen.

4.3.2 Potrebujemo

- Raspberry Pi z opremo
 - LED diodo
 - upor
 - 2 žici, ki imata na eni strani ženski in na drugi moški priključek
 - povezovalno ploščo

Namesto žensko-moške žice lahko uporabimo eno žensko in eno moško žico in ju povežemo. Na sliki 4.3 so prikazani pripomočki za vklop led diode.



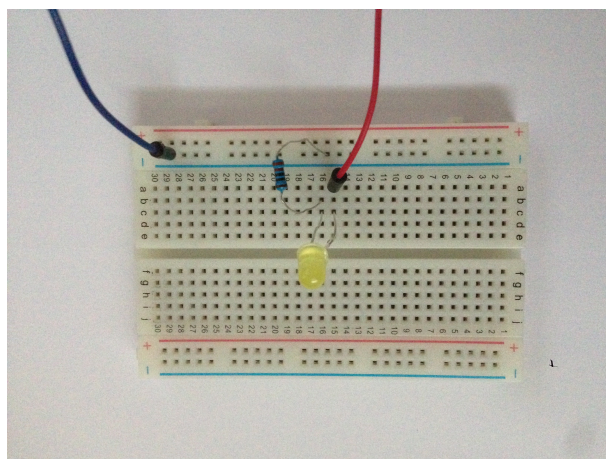
Slika 4.4: Povezovalna plošča.

Na Raspberry Pi pred začetkom učne ure namestimo tudi ScratchGPIO. Navodilo za namestitev najdemo v poglavju 3.1 ScratchGPIO.

4.3.3 Fizični del

Otrokom pokažemo povezovalno ploščo, LED diodo, upor in žice. Naloga je prižgati LED diodo. Opazimo, da ima dioda dve nogici, eno krajšo in eno daljšo. Če želimo, da dioda posveti, moramo skozi daljšo nogico poslati električni tok, ki potem nadaljuje pot skozi krajšo. To je otrokom razumljivejša, poenostavljena razlaga dogajanja. Tok pošljemo preko pinov na plošči Raspberry Pi. Pin in LED diodo preko povezovalne plošče povežemo z žico. Otrokom pokažemo, kako so povezane vrstice in kako stolpci na povezovalni plošči 4.4.

Krajšo izmed nogic LED diode vstavimo v poljubno luknjo stolpca označenega z modro barvo in minusom. Moder stolpec nato povežemo z ozemljitvenim pinom. Daljšo nogico LED diode vstavimo v poljubno vrstico in v isto vrstico vstavimo tudi eno izmed nogic upora. Drugo nogico upora vstavimo v eno izmed preostalih vrstic bodisi na isti, bodisi na drugi strani povezovalne plošče. Vrstico, v kateri je druga nogica upora, nato povežemo z GPIO



Slika 4.5: Vezava LED diode.

pinom. Takšna vezava je predstavljena na sliki 4.5.

Zdaj pa si pogledjmo, na katere pine Raspberry Pija povežemo žici. Raspberry Pi prve generacije model B+ ima 40 pinov, ki so predstavljeni na sliki 4.6. Za druge modele si je potrebno razporeditev pinov pogledati v njihovi dokumentaciji. Pini so označeni s številkami od 1 do 40, ob njih pa je napisana njihova uporaba. S črno barvo in napisom GND so označeni ozemljitveni pini. Izberemo si torej enega izmed pinov 6, 9, 14, 20, 25, 30, 34 ali 39 in nanj povežemo žico, ki povezuje stolpec povezovalne plošče označen z modro barvo in minusom. Pina, označena z belo barvo in napisom DNC, kar je kratica za “Do Not Connect” ali po slovensko “ne priklaplaj”, se ne uporabljata. Pini, označeni z napisom GPIO, predstavljajo vhodne in izhodne pine Pina številka 2 in 4 imata konstantno napetost 5 V. Pina številka 1 in 17 imata konstantno napetost 3.3 V. Z njima lahko preverimo, ali smo LED diodo pravilno povezali. Če žico, ki je povezana z daljšo nogico LED diode, povežemo s pinom 1 ali 17, LED dioda sveti. Z otroki ugotovimo, ali lahko brez premikanja elementov nadziramo, kdaj LED dioda sveti in kdaj ne.

Za tem moramo ugotoviti, ali potrebujemo vhod ali izhod. To je odvisno od tega, ali želimo računalniku podajati informacije iz LED diode, ali pa želimo, da računalnik podaja informacije LED diodi. Prav tako je potrebno

	Pin No.		
3.3V	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3.3V	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7
DNC	27	28	DNC
GPIO5	29	30	GND
GPIO6	31	32	GPIO12
GPIO13	33	34	GND
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
GND	39	40	GPIO21

Slika 4.6: Pini Raspberry Pi prve generacije modela B+ [11].

razmisliti, ali je računalniška miška vhodna ali izhodna naprava. Enako velja ugotoviti za tipkovnica, ekran, zvočnike, tiskalnik in mikrofona? Naša želja je, da računalnik preko pina diodo prižge ali ugasne, kar pomeni, da je računalnik tisti, ki podaja informacije in zato potrebujemo izhodni pin. Vsi GPIO pini so lahko tako vhodni kot izhodni. Izberemo si na primeri pin 11, ki ima oznako GPIO in žico, ki je preko upora povezana z daljšo nogico LED diode, priklopimo nanj.

Zdaj je čas, da Raspberry Pi vklopimo. Prikazalo se nam bo namizje. Zaženemo aplikacijo ScratchGPIO.

4.3.4 Programski del

Oglejmo si aplikacijo. V levem zgornjem delu najdemo osem skupin ukazov: motion - premikanje, looks - izgled, sound - zvok, pen - svinčnik, control - krmiljenje, sensing - zaznavanje, operators - operatorji, variables - spre-



Slika 4.7: Ukaz, s katerim prižgemo LED diodo, priključeno na pin 11.

menljivke. Za prižig LED diode bomo potrebovali skupino krmiljenje, ki je označena z oranžno barvo. Kliknemo nanjo in si oglejmo ukaze te skupine. Z otroki prevedemo ukaze v slovenščino in jih skušamo pojasniti.

Naša naloga je prižgati LED diodo in jo za tem tudi ugasniti. LED diodo smo priključili na pin 11. Da se LED dioda prižge, mora Raspberry Pi pin 11 prižgati. To naredimo z ukazom “broadcast”, ki ga najdemo v skupini krmiljenje. Ukaz z miško povlečemo na veliko prazno polje na sredini aplikacije. Ukaz potrebuje argument. Vidimo, da se desno od napisa “broadcast” nahaja polje s puščico. Če nanjo kliknemo, se pojavi spustni meni, v katerem je takrat zgolj možnost “new”. Kliknemo na to izbiro in pojavilo se bo okno, v katero vpišemo nov argument. Napišemo “Pin11On” 4.7 in potrdimo.

4.3.5 Zaporedje ukazov

Dodajmo še ukaz, ki ob kliku zastavice sproži ukaze, ki mu sledijo. Tako ob kliku zastavice pin 11 prižgemo, kar prižge našo LED diodo. Ampak LED dioda je zdaj ves čas prižgana. Naredimo tako, da se LED dioda prižge in nato ugasne. Razmislimo, kateri ukaz uporabimo, da jo ugasnemo. Če “on” prižge pin, ga “off” ugasne. Iz nabora ukazov na levi povlečemo še en broadcast, ga sklenemo z doslej zapisanimi ukazi in tokrat kot argument zapišemo “Pin11Off”. V programskem okolju ScratchGPIO ukaze sestavljamo kot lego kocke in tako sestavimo zaporedje ukazov. Ukazi se izvršujejo eden za drugim. Ob kliku zastavice se LED dioda za trenutek prižge, vendar se takoj spet ugasne. Ukazi se izvršujejo drug za drugim, vendar tako hitro, da verjetno sploh ne opazimo, da se je dioda prižgala. To rešimo tako, da Raspberry Piju naročimo, da vmes malo počaka. Ukaz, ki nam pri tem pomaga, se



Slika 4.8: Prižig ene LED diode.



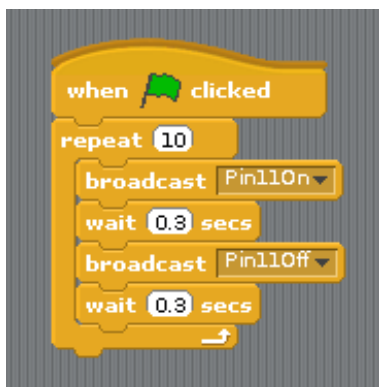
Slika 4.9: Utripanje LED diode.

imenuje “wait” in sprejme argument, koliko sekund naj čaka. Na sliki 4.8 je zaporedje ukazov, kjer je čas čakanja in s tem prižgane LED diode 1 s.

4.3.6 Ponavljalna zanka

Naslednja naloga je, da LED dioda utripa. Uporabili bomo ukaz “repeat”, s katerim bomo ponavljali prižiganje in ugašanje LED diode. Razmislimo, ali je zaporedje ukazov na sliki 4.9 pravilno.

Kaj se bo zgodilo po prvi iteraciji? LED dioda se bo prižgala takoj, ko se bo ugasnila. Med ukazom za izklop in ukazom za vklop bo potrebno vstaviti ukaz za čakanje. Lahko ga postavimo bodisi za broadcast “Pin11Off”, bodisi pred broadcast “Pin11On”. Razmislimo, kaj moramo spremeniti, če želimo hitrejšo utripanje. Primer končne rešitve je predstavljen na sliki 4.10.



Slika 4.10: Pravilno utripanje LED diode.

4.4 Novoletna smreka

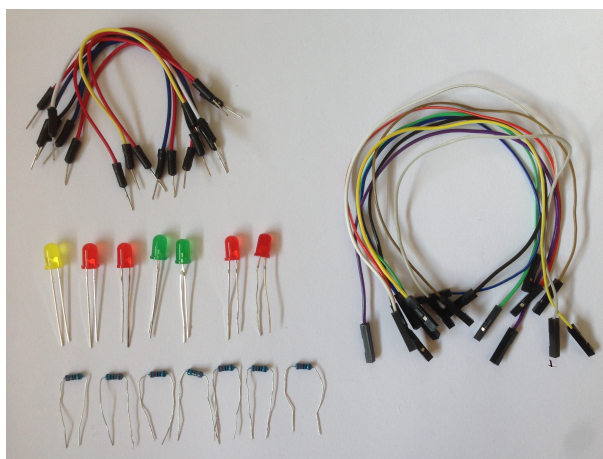
4.4.1 Učni cilji

Tokrat se bomo naučili, kako priklopiti in prižgati več LED diod. Izdelali bomo novoletno smreko, ki jo bomo okrasili z LED diodami. Ustvarili bomo več različnih načinov utripanja diod. Utrdili bomo znanja o zaporedju ukazov in ponavljalnih zankah.

4.4.2 Potrebujemo

- Raspberry Pi z opremo
 - povezovalno ploščo
 - 7 LED diod različnih barv
 - 7 uporov
 - 9 žensko-moških žic
- papir
- barvice
- šivanko

Potrebščine so prikazane na sliki 4.11.



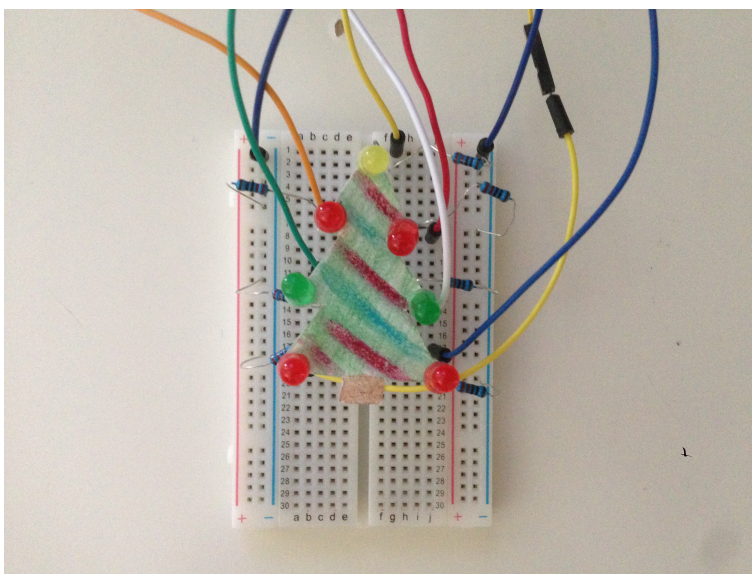
Slika 4.11: Potrebščine za izdelavo novoletne smreke.

4.4.3 Fizični del

Otroci naj na papir narišejo smrečico in jo izrežejo. Najlažje je, če je smrečica široka toliko, kot je širok notranji del povezovalne plošče. Smrečico naj tudi pobarvajo. Sledi okrasitev z LED diodami. Najbolj enostavno je najprej v papir narediti majhno luknjico s šivanko in nato skozi luknjico vstaviti eno izmed nogic LED diode. Če luknjico naredimo kar z LED diodo, se lahko papir raztrga ali pa nogico LED diode ukrivimo.

LED diode z upori in žicami povežemo s pini Raspberry Pijem kot smo se naučili pri prejšnji nalogi. Vsaka dioda mora biti z eno žico preko upora povezana na GPIO in z drugo na ozemljitveni pin. Končen izdelek je prikazan na sliki 4.12.

Zapišemo si, na kateri GPIO pin smo priključili katero LED diodo. Na primer, najvišja LED dioda je priključena na pin številka 7, dioda levo pod njo je priključena na pin 11, njena soseda na 12, diodi 3. nivoja sta na pinih 15 in 16 in zadnji dve na 21 in 22.



Slika 4.12: Novoletna smreka.

4.4.4 Programski del

Naloga je zaporedoma prižigati in ugašati LED diode. Najprej naj se prižge in ugasne najvišja dioda in nato dioda levo pod njo, za njo se prižge njena desna sosedica, sledi dioda levo pod njo in tako naprej do najnižje desne diode.

Nastala je zelo dolga kača ukazov, ki jo prikazuje slika 4.13.

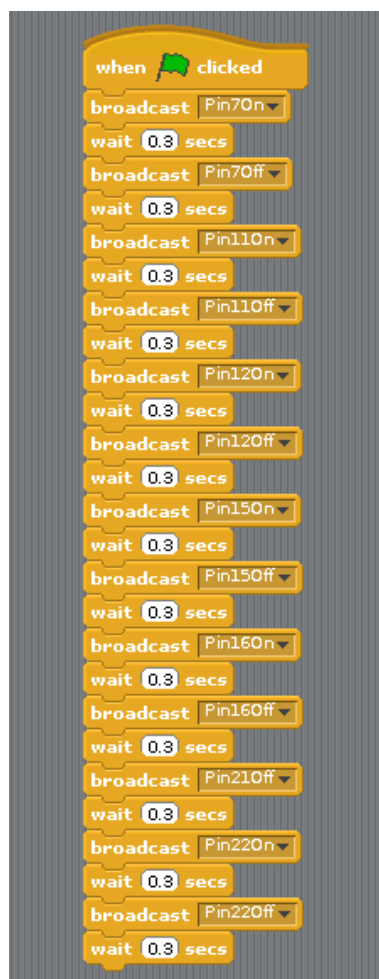
Naredimo nalogo še tako, da se najprej prižge in ugasne LED dioda 1. nivoja, nato se hkrati prižgreta in ugasneta diodi 2. nivoja in tako naprej do najnižjega nivoja. Ukazi za takšno prižiganje so prikazani na sliki 4.14.

Otroci naj si izmislijo svoj vzorec prižiganj in ugašanj in ga realizirajo.

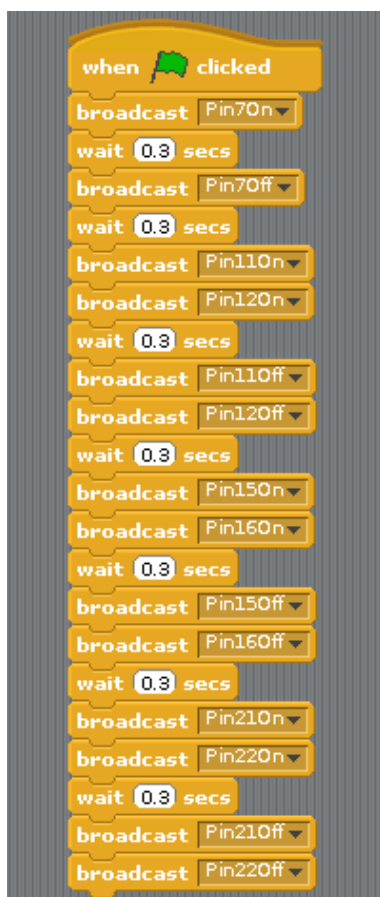
4.5 Policijski avtomobil

4.5.1 Učni cilji

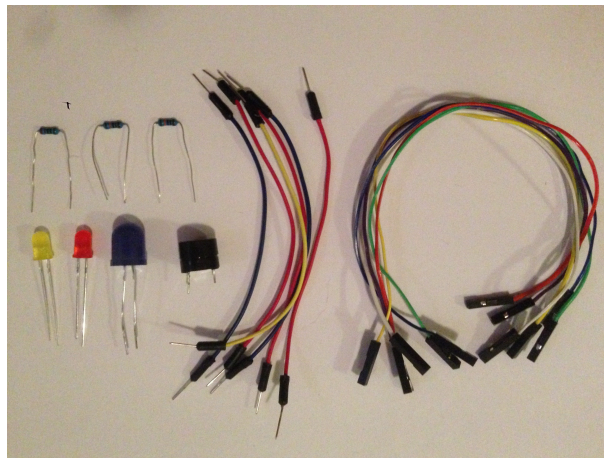
Izdelali bomo policijski avtomobil s policijsko sireno in avtomobilskimi lučmi. Pri tej nalogi otroci spoznajo nov element, piskač. Utrdijo svoje znanje prižiganja diod in izboljšajo razumevanje ponavljalnih zank.



Slika 4.13: Prižiganje in ugašanje LED diod novoletne smreke.



Slika 4.14: Prižiganje in ugašanje LED diod po nivojih.



Slika 4.15: Potrebščine za izdelavo policijskega avtomobila.

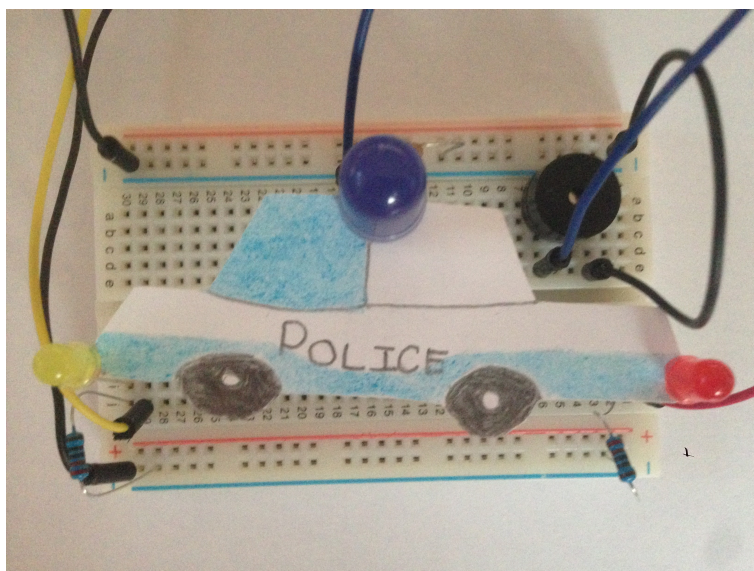
4.5.2 Potrebščine

- Raspberry Pi z opremo,
 - povezovalno ploščo,
 - LED diodo rumene barve,
 - LED diodo rdeče barve,
 - LED diodo modre barve
- 3 upore
- piskač
- 5 žensko-moških žic
- 1 moško žico
- papir
- barvice
- šivanko

Potrebščine za izdelavo policijskega avtomobila so prikazane na sliki 4.15.

4.5.3 Fizični del

Otroci naj na papir narišejo policijski avto, ga izrežejo in pobarvajo. Nato naj avtu dodajo sprednjo in zadnjo luč ter policijsko luč. Priporočljiva je



Slika 4.16: Policijski avtomobil.

uporaba šivanke za prebadanje papirja. Na povezovalno ploščo dodamo še piskač, katerega daljšo nogico povežemo z GPIO pinom, krajšo pa z moško žico povežemo s stolpcem, označenim z modro barvo. Izdelek je prikazan na sliki 4.16.

Zapomnimo si, v kateri pin vstavimo katero žico. Rdečo vzvratno luč priključimo na primer na pin 11. Sprednjo, rumeno luč, na pin 13, policijsko, modro, pa na pin 15. Piskač povežimo s pinom 12.

4.5.4 Programski del

Sprednja luč avtomobila mora biti po predpisih ves čas vožnje prižgana. Zato naj bo rumena LED dioda ves čas prižgana. Po nekem času prižgimo policijsko luč, ki mora hitro utripati. Hkrati z lučjo pa se mora vklopiti tudi sirena. Naša modra LED dioda se mora ponavljajoče vklapljati in izklapljati. Ob prižiganju in ugašanju diode prižigamo in ugašamo tudi piskača. Ko se policijska sirena zaključí, prižgemo še rdečo LED diodo.

Otrokom povemo, da piskač nadzorujemo enako kot LED diode - s prižiganjem



Slika 4.17: Prižiganje luči in policijske sirene.

in ugašanjem pinov.

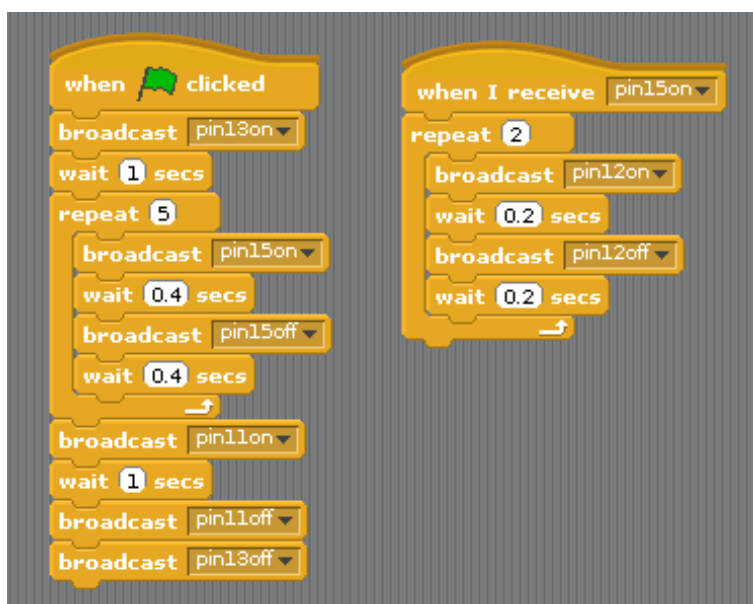
Možna rešitev je predstavljena na sliki 4.17.

Kot dodatno nalogo lahko otroci sestavijo ukaze, po katerih naj piskač piska hitreje kot LED dioda utripa. Primer rešitve je predstavljen na sliki 4.18.

4.6 Glasbeni inštrument

4.6.1 Učni cilji

Ustvarili bomo glasbeni inštrument z nekaj tipkami, ki bodo ob pritisku predvajali različne zvoke. Pri tej nalogi se otroci spoznajo s pogojnimi stavki. Naučijo se uporabljati tipke in posledično tudi uporabe GPIO pinov kot vhodov. Novo je tudi predvajanje zvoka v Scratchu in uporaba ukazov skupine za zaznavanje.



Slika 4.18: Piskač piska hitreje kot LED dioda utripa.

4.6.2 Potrebujemo

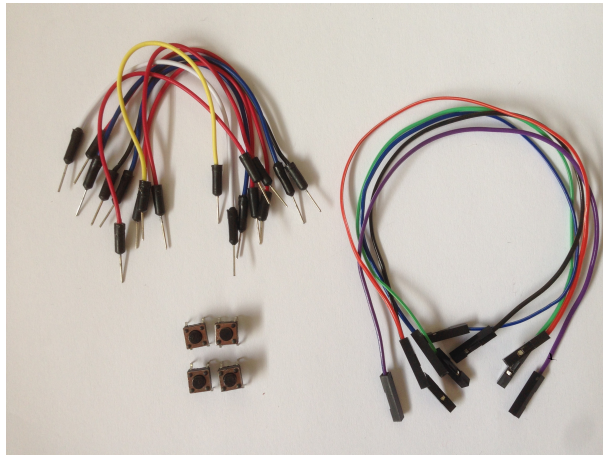
- Raspberry Pi z opremo,
 - povezovalno ploščo,
 - 4 tipke,
 - 5 žensko-moških žic,
 - 4 moške žice

Potrebščine za izdelavo glasbenega inštrumenta so prikazane na sliki 4.19.

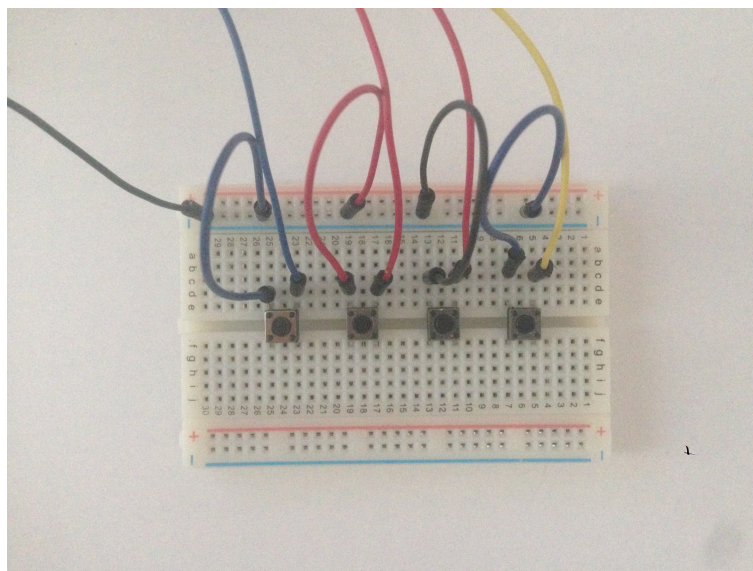
4.6.3 Fizični del

Tipke razporedimo po povezovalni plošči. Eno izmed nogic povežemo z modrim stolpcem, drugo pa z GPIO pinom. Moder stolpec povežemo z ozemljitvenim pinom. Glasbeni inštrument je prikazan na sliki 4.20.

Ko smo uporabljali LED diode in piskača, smo pin uporabili kot izhod. Z otroki razmislimo, kako bo s tem pri tipkah in ugotovimo, da bomo pine uporabili kot vhode. Pomembno je, da ne uporabimo GPIO pinov, ki smo



Slika 4.19: Potrebščine za izdelavo glasbenega inštrumenta.



Slika 4.20: Glasbeni inštrument.

jih pred tem uporabili za izhod. Najbolje bo, da Raspberry Pi ugasnemo in ponovno zaženemo.

4.6.4 Programski del

Uporabimo GPIO pine 40, 38, 36 in 32. Naša naloga je, da ob pritisku na tipko instrument predvaja določen zvok. Uporabili bomo ukaz "if", s pomočjo katerega oblikujemo pogojni stavek. Z otroki razmislimo, ob katerem pogoju želimo, da se predvaja zvok in ugotovimo, da je pogoj pritisnjena tipka. Ukaz "if", ki ga najdemo med ukazi za krmiljenje, ima okvirček, v katerega vstavimo pogoj, ob katerem se izvršijo ukazi znotraj ukazne zanke. Kadar je tipka pritisnjena, je vrednost senzorja pina, na katerega je priključena, enaka nič. V skupini ukazov "sensing" oziroma zaznavanje izberemo blok "sensor value", ki na začetku sprejme argument. Izberemo pin 40, na katerega je priključena prva tipka. Nato v skupini ukazov (operators) poiščemo znak "=". Moder ukaz pin40 "sensor value" premaknemo na levo stran enačaja, na desno pa vpišemo število 0. To je pogoj, ob katerem želimo predvajati zvok. V skupini ukazov "Sound" oziroma zvok poiščimo poljuben ukaz za predvajanje zvoka. Izberemo lahko na primer "play drum" ali "play sound meow" ali "play note". Argumente ukaza "play drum" lahko poljubno spreminjamo kot vidimo na sliki 4.21.

Otroci naj sami zapišejo ukaze še za preostale tri tipke. Vsaka tipka naj predvaja drug zvok. Ob tem moramo z otroki ugotoviti, kam postavljamo pogojne stavke - enega v drugega ali enega za drugega. Prav tako je potrebno razmisliti, ali je dovolj, da samo enkrat preverimo, ali je tipka pritisnjena. Uporabimo zanko "forever", ki bo vedno izvajala ukaze znotraj nje. Najdemo jo v skupini ukazov za krmiljenje.

Primer končne rešitve vidimo na sliki 4.22.



Slika 4.21: Različne možnosti ukaza "play drum".

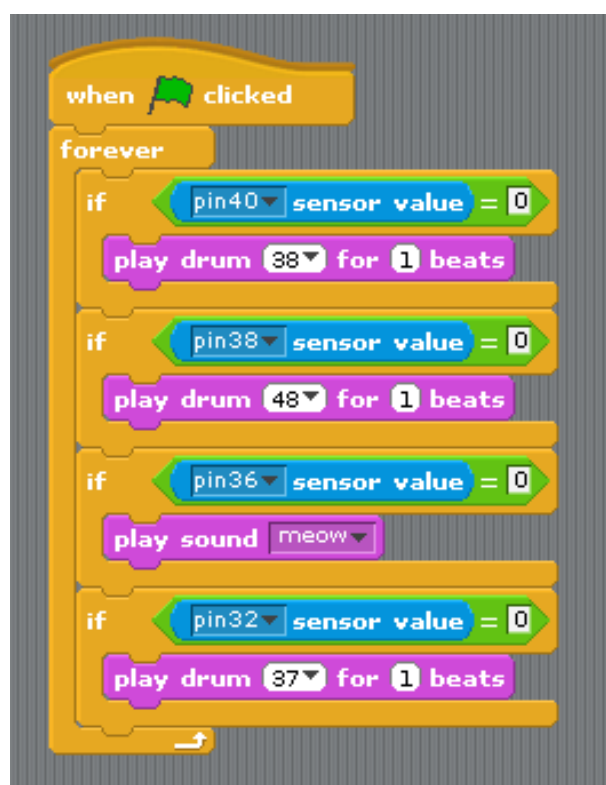
4.7 Robotova roka

4.7.1 Učni cilji

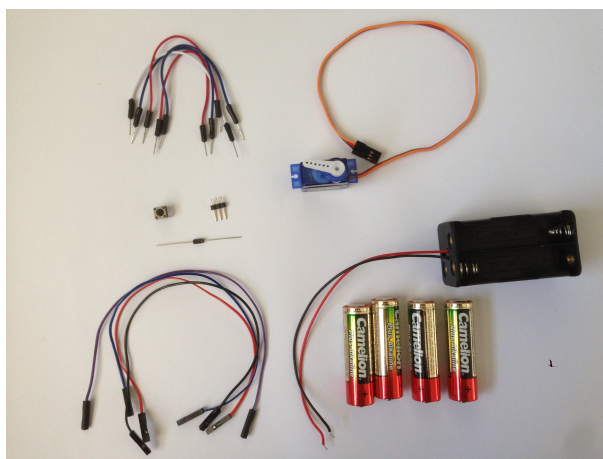
Izdelali bomo robotka, ki maha z roko. Otroci se spoznajo z mikro servomotorjem in se naučijo, kako ga priključiti in nato uporabiti. S pomočjo te naloge otroci utrdijo znanje matematike na področju kotov.

4.7.2 Potrebujemo

- Raspberry Pi z opremo,
 - povezovalno ploščo,
 - mikro servomotor,
 - pin header,
 - diodo,
 - 4 žensko-moški žici,
 - 1 moško žico,



Slika 4.22: Primer končne rešitve glasbenega inštrumenta.



Slika 4.23: Potrebščine za izdelavo robotove roke.

- papir,
- barvice
- 1 gumb.

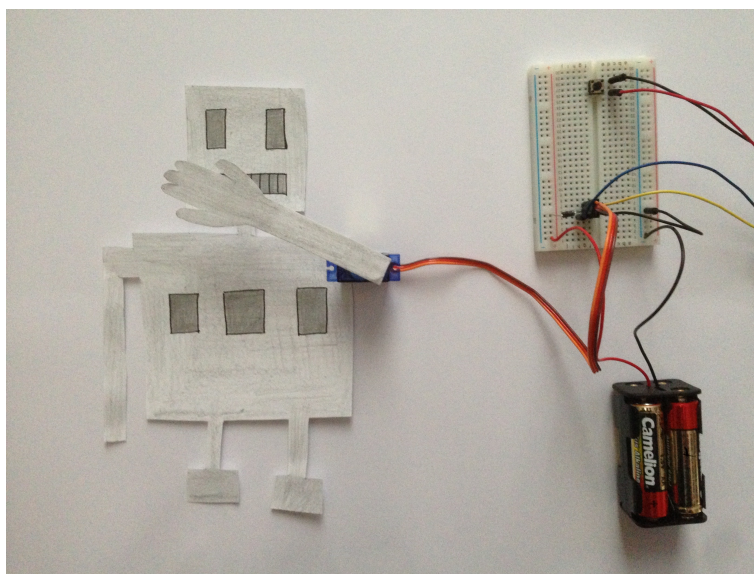
Potrebščine za izdelavo robotove roke so prikazane na sliki 4.23.

4.7.3 Fizični del

Otroci naj iz papirja izdelajo robotka, nanj prilepijo mikro servomotor in na njegovo ročico prilepijo robotkovo roko. Servo s pomočjo pin headerja vtaknemo v tri različne vrstice breadboarda. Žice servomotorčka so treh različnih barv. Prva je rjava ali črna in povežemo jo z ozemljitvenim pinom Raspberry Pija ter z negativnim delom baterijskega napajalnika. Srednja žica je rdeče barve. Povežemo jo preko diode z pozitivnim delom baterijskega napajalnika. Zadnja žica pa je rumene ali oranžne barve. Povežemo jo z GPIO pinom 11. Končni izdelek si lahko pogledamo na sliki 4.24.

4.7.4 Programski del

Mikro servomotor upravljamo tako, da ustvarimo spremenljivko servo in številko pina, na katerega je servo priključen, v našem primeru "servo11".



Slika 4.24: Robotova roka.



Slika 4.25: "Nastavljanje lege ročice servomotorja".

Spremenljivko ustvarimo v skupini ukazov "variables". Mikro servomotor postavimo na njegovo naravno lego, ki je 0 stopinj. To storimo tako, da med ukazi spremenljivk poiščemo "set _ to _", ki sprejme dva argumenta. V prvega vstavimo "servo11", saj želimo premakniti servo, ki je na pinu 11. Drugi argument pa nam pove, v katero lego želimo postaviti servo11 4.25.

Otroci naj poskusijo število 0 spremeniti v poljubno število med 0 in 90. Ugotovijo lahko tudi, kaj se zgodi, če bi pred to številko zapisali znak minus. Servomotor premakne ročko v nasprotno smer. Servomotor lahko ročko premika med -90 in 90 stopinjami. Razmislimo, za koliko ga lahko največ premaknemo. Želeli bi, da naš robot maha kolikor je mogoče. Naredimo program, ki bo robotovo roko trikrat zaporedoma premaknil od leve proti



Slika 4.26: Možna rešitev Robotove roke.

desni in nazaj. Možna rešitev je prikazana na sliki 4.26.

4.7.5 Dodatno

Robot naj pomaha, ko pritisnemo na gumb. Se še spomnimo, kako smo uporabljali gumbe pri glasbeni skrinjici?

Možna rešitev je prikazana na sliki 4.27.

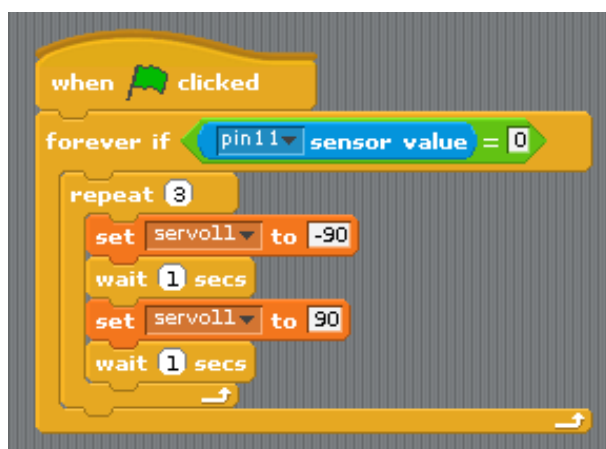
4.8 Twister

4.8.1 Učni cilji

Ustvarili bomo pripomoček za igranje otrokom zelo priljubljene igre twister. Otroci se naučijo, kako narediti naključno izbiro z uporabo ukaza "random". Malo bolje spoznajo tudi kote.

4.8.2 Potrebujemo

- Raspberry Pi z opremo,
 - 2 povezovalni plošči,
 - micro servomotor,
 - diodo,



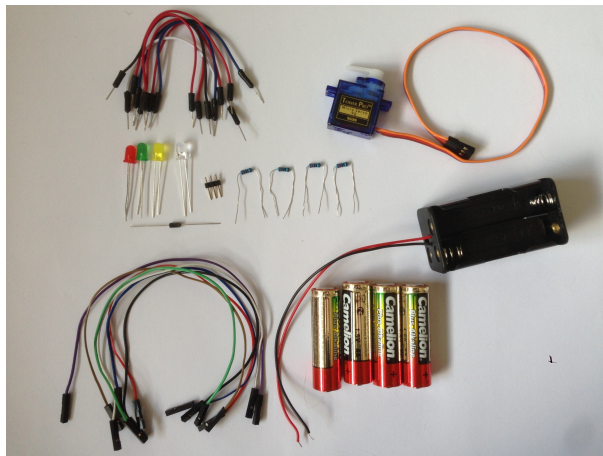
Slika 4.27: Robot pomaha, ko pritisnemo na gumb.

- 4 led diode različnih barv,
- 4 upore,
- 7 žensko-moških žic,
- 1 moško žico,
- baterijski napajalnik ,
- papir,
- barvice.

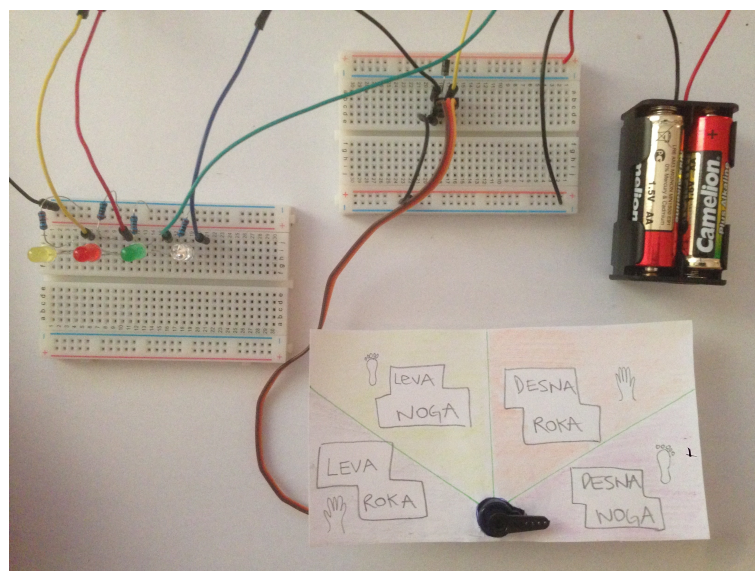
Potrebščine za izdelavo pripomoča za igro Twister so prikazane na sliki 4.28.

4.8.3 Fizični del

Na eno povezovalno ploščo naj otroci vstavijo 4 led diode in jih pravilno povežejo z Raspberry Pijem kot smo se učili v prejših učnih urah. Na drugo povezovalno ploščo pa povežejo servomotor kot smo ga povezali pri nalogi Robotova roka. Iz papirja naj otroci izrežejo pravokotnik, ki mu na sredini spodaj izrežejo luknjo za servomotor. Nato naj pravokotnik razdelijo na 4 dele. Vsak del naj pobarvajo z drugo barvo in na njih zapišejo leva roka, desna roka, desna noga in leva noga. Nato naj list prilepijo na servomotor kot kaže slika 4.29.



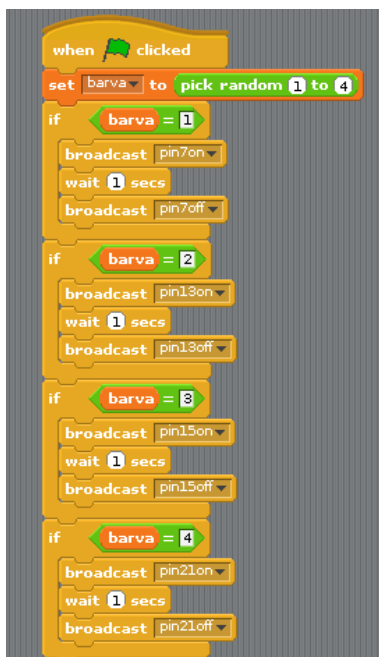
Slika 4.28: Potrebščine za izdelavo pripomoča za igro Twister.



Slika 4.29: Pripomoček za igro Twister.



Slika 4.30: Naključna izbira.



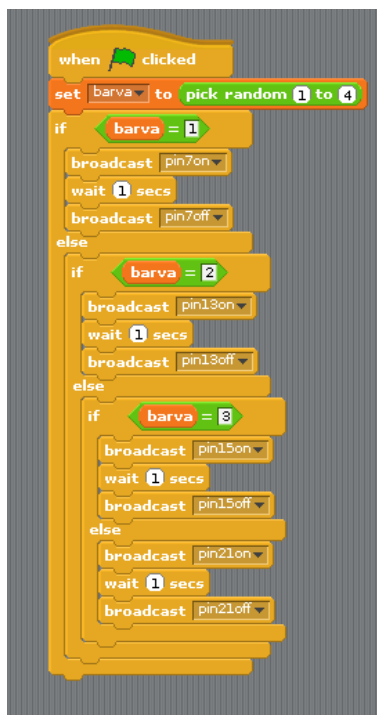
Slika 4.31: Program za naključni prižig LED diode.

4.8.4 Programski del

Lotimo se najprej LED diod. Želimo, da se vsakič zasveti naključno izbrana LED dioda. Raspberry Pi naj si izbere naključno številko med 1 in 4. Ustvarimo spremenljivko “barva”, ki bo hranila vrednost naključno izbranega števila. Za izbiro naključnega števila uporabimo ukaz “pick random” iz skupine ukazov “operators” 4.30.

Program sestavimo tako, da bo v primeru izbora števila 1 posvetila prva dioda, pri izbiri števila 2 bo posvetila druga dioda in tako dalje.

Rešitev na sliki 4.31 je pravilna, vendar pa jo je možno izboljšati. Predpostavimo, da je naključno izbrano število 2. Najprej program preveri, ali je

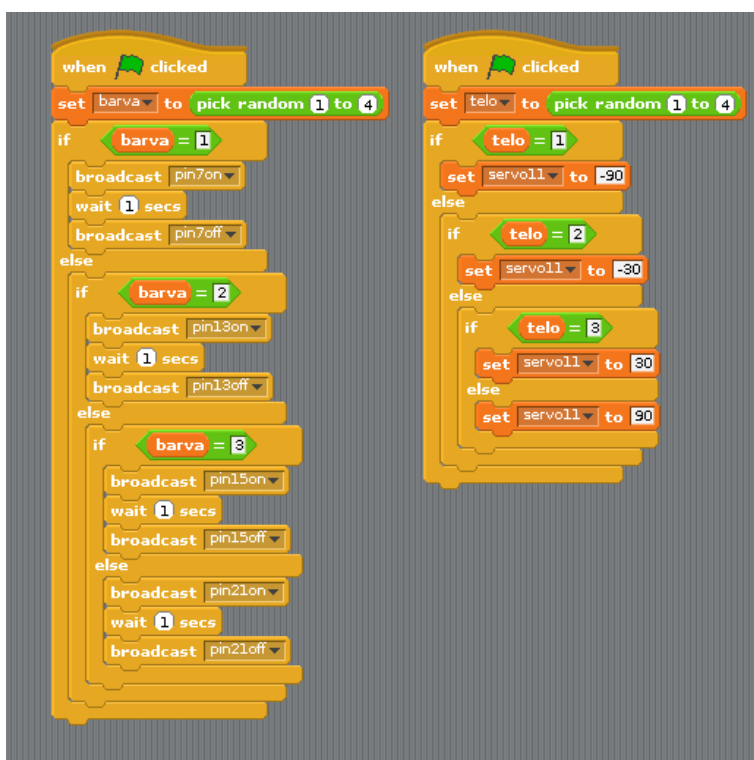


Slika 4.32: Izboljšan program za naključni prižig LED diode.

izbrano število 1. Ko ugotovi, da temu ni tako, preveri, ali je izbrano število 2. Pogoji so resnični, zato se druga LED dioda prižge in ugasne. Na tem mestu bi lahko zaključili, saj je naloga opravljena, vendar pa programa še ni konec, saj le-ta še preverja, ali sta izbrani števili 3 in 4. Temu se lahko izognemo z uporabo pogojnega stavka “if, else”. Tako dosežemo, da program preveri, če je izbrana številka 1 in v tem primeru prižge prvo LED diodo, v nasprotnem primeru pa preveri, ali je izbrana številka 2 in tako dalje. Ta del naj otroci skušajo narediti sami.

Primer rešitve vidimo na sliki 4.32.

Podobno kot z LED diodami moramo zdaj narediti še s servomotorjem. Tudi tu potrebujemo naključno izbrano število in nato premakniti ročico servomotorja glede na to število. Najprej izračunamo, kam naj kaže ročica servomotorja. Najbolje je, da na tablo narišemo kopijo lista, ki je pritrjen na servomotor. Imamo 4 polja; v prvem polju kaže servo na -90, v zadnjem na



Slika 4.33: Končna rešitev naloge Twister.

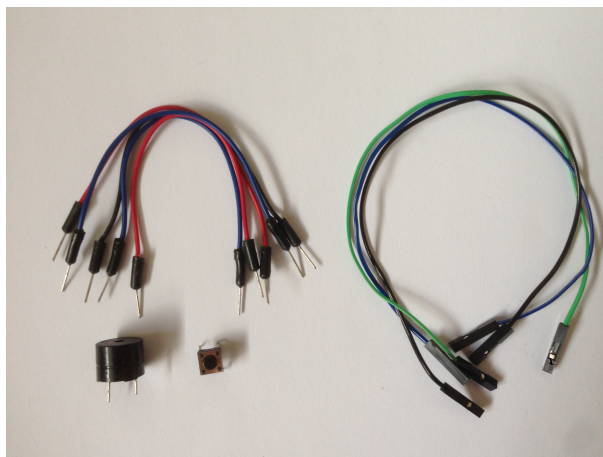
90. Da ročica servomotorja pokaže na drugo polje, ga postavimo na -30, da pokaže tretje pa na 30. Ustvarimo torej novo spremenljivko, ki ji pripišemo naključno vrednost med 1 in 4. Nadaljevanje pa naj otroci naredijo sami po vzoru prvega dela naloge z barvami.

Primer končne rešitve vidimo na sliki 4.33.

4.9 Budilka

4.9.1 Učni cilji

Izdelali bomo preprosto budilko. Preko naloge otroci utrdijo znanje pridobljeno v prejšnjih nalogah. Spoznajo ukaze povezane s časom in se naučijo prekiniti program.



Slika 4.34: Potrebščine za izdelavo budilke.

4.9.2 Potrebujemo

- Raspberry Pi z opremo,
 - povezovalno ploščo,
 - tipko,
 - piskač,
 - 3 žensko-moške žice,
 - 2 moški žici.

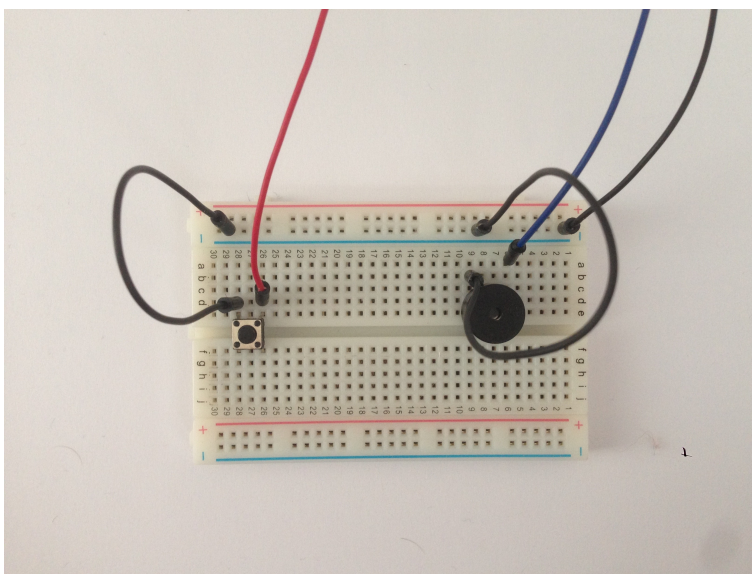
Potrebščine za izdelavo budilke so prikazane na sliki 4.34.

4.9.3 Fizični del

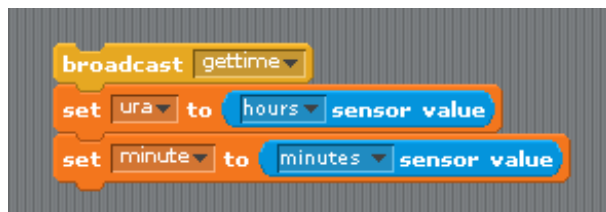
V povezovalno ploščo vstavimo tipko ter piskač, ju povežemo z modrim stolpcem, ki ga priključimo na ozemljitveni pin Raspberry Pija. Tipko in piskač povežemo tudi z GPIO pinoma, kot je prikazano na sliki 4.35.

4.9.4 Programski del

Če ukazu “broadcast” kot argument napišemo “gettime”, dobimo z ukazom “sensor value”, ki mu kot argument izberemo “hours”, trenutno uro, z argumentom “minutes” pa trenutne minute. Otroke vzpodbudimo, da to pre-



Slika 4.35: Budilka.

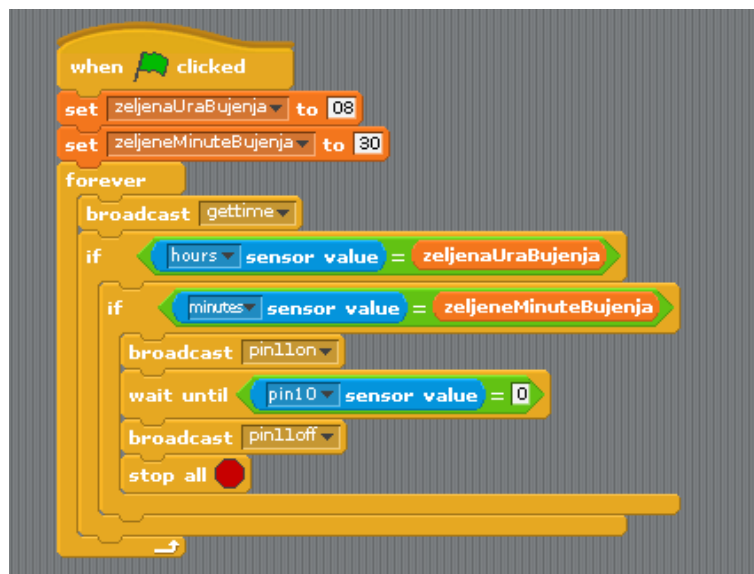


Slika 4.36: Pridobivanje trenutnega časa.

izkusimo. Ustvarimo spremenljivki *ura* in *minute* ter jima določimo vrednost kot vidimo na sliki 4.36.

Desno zgoraj v okencu z mačkom se bo pojavilo ime spremenljivke in njena vrednost. Preverimo, ali kaže pravilen trenutni čas.

Zdaj lahko naredimo budilko. Najprej moramo nastaviti uro, ob kateri se želimo zbuditi. Ustvarili bomo spremenljivki “*zeljenaUraBujenja*” in “*zeljeneMinuteBujenja*”. Z ukazom “*set*” jima določimo željeno vrednost. Nato pa je potrebno preveriti, ali je trenutna ura enaka željeni uri bujenja ter v primeru ujemanja preveriti, ali se tudi minute ujemajo. Če se ujemajo tako ure kot tudi minute, sprožimo piskača. Če želimo preizkusiti, ali naloga de-



Slika 4.37: Primer končne rešitve naloge budilka.

luje pravilno, lahko željeno uro bujenja in željene minute bujenja nastavimo na trenutni čas.

Alarm bo potrebno tudi izklopiti. Želimo, da se ob pritisku na tipko piskanje konča. Uporabimo lahko ukaz “wait until”, ki sprejme časovni pogoj, po katerem izvede določen ukaz. V našem primeru naj torej ob pritisku gumba sporoči piskaču, da se ugasne. Z otroki stestiramo delovanje. Preverimo, ali piskač neha piskati ob pritisku na gumb. Če piskač še naprej piska, smo pozabili, da se zanka ves čas ponavlja. Raspberry Pi je ponovno preveril, ali se trenutni čas ujema z izbranim, prišel do ugotovitve, da je temu res tako, ter ponovno sprožil piskača. Potrebno mu je torej naročiti, da ob pritisku na tipko ne želimo več preverjati ujemanja časa. Eden izmed načinov je uporaba ukaza “stop all”, ki prekine izvajanje programa”.

Primer končne rešitve vidimo na sliki 4.37.

Poglavje 5

Sklepne ugotovitve

V diplomski nalogi smo najprej spoznali mini računalnik Raspberry Pi, čemu je namenjen, kako je sestavljen, katere operacijske sisteme lahko uporabimo in predstavili nekaj primerov uporabe. Opisali smo tudi postopek prve uporabe Raspberry Pija. Sledila je predstavitev programskega jezika in okolja za učenje programiranja, Scratch, ter njegove verzije za uporabo pinov, ScratchGPIO.

V nadaljevanju so predstavljene naloge, namenjene staršem, učiteljem, starejšim bratom, sestram in drugim, ki bi želeli učiti otroke osnov programiranja. Najprej so predstavljeni osnovni elementi, s katerimi se srečamo pri aktivnostih. Nato sledijo učne ure, ki se pričnejo z uvodom, kjer otroci spoznajo mini računalnik Raspberry Pi. Naloge so namenjene otrokom starim med 8 in 13 let. Izkušnje z učenjem programiranja za osnovnošolske otroke sem pridobila z izvajanjem interesne dejavnosti na Osnovni šoli Kette in Murn. Prišla sem do ugotovitve, da nekateri otroci še ne znajo hitro tipkati, zato je poučevanje s pomočjo programskega jezika Scratch veliko lažje kot s pomočjo programskega jezika Python. Scratch je otrokom zelo všeč, saj je interaktiven, hitro razumljiv in pregleden. Aktivnosti, predstavljene v diplomski nalogi so preizkušene z otroki starimi med 8 in 13 let.

Najpomembnejši izziv je bil z vsako nalogo ustvariti zgodbo, ki bo otroke pritegnila in jim popestrila učenje programiranja. Naloge dodajo smisel

prižiganju LED diod in premikanju servomotroja. Tako na primer je LED diode potrebno prižgati, da okrasimo novoletno smrečico, servomotor pa premikamo, da robot pomaha. Otrokom je všeč, da lahko pri nalogah uporabijo tudi svojo domišljijo.

Literatura

- [1] Richardson M. in Wallace S., Getting Started with Raspberry Pi. 1. izd. Sebastopol: Maker Media, Inc., 2012. ISBN 978-1-4493-4421-4.
- [2] Upton, E. in Halfacree, G., Meet the Raspberry Pi. John Wiley & Sons, 2012. ISBN 9781118494684
- [3] Hoile C. et al., Make: Raspberry Pi and AVR Projects: Augmenting the Pi's ARM with the Atmel ATmega, ICs, and Sensors. Maker Media, Inc., 2014. ISBN 9781457186202
- [4] Andrews, C., Easy as pi. Engineering Technology, 2013, št. 3, str. 34-37.
- [5] Cymplecy (SimpleSi)[Online]. Dosegljivo:
<http://simpleSi.net/scratchgpio/scratch-raspberrypi-gpio/>.
- [6] SparkFun Electronics[Online]. Dosegljivo:
<https://learn.sparkfun.com/>.
- [7] Raspberry Pi GPIO home automation[Online]. Dosegljivo:
<http://www.instructables.com/id/Raspberry-Pi-GPIO-home-automation/>. [Dostopano 28. 7. 2015]
- [8] Raspberry Pi Foundation[Online]. Dosegljivo:
<https://www.raspberrypi.org/>.
- [9] Slovenska Raspberry Pi skupnost[Online]. Dosegljivo:
<http://slo-pi.com/>.

- [10] Auto Login and Auto Start in Raspberry Pi[Online]. Dosegljivo:
<http://www.opentechguides.com/how-to/article/raspberry-pi/5/raspberry-pi-auto-start.html>.
- [11] GPIO Pins[Online]. Dosegljivo:
<http://raspi.tv/wp-content/uploads/2014/07/Raspberry-Pi-GPIO-pinouts.png>.
- [12] C. Severance. Eben upton: Raspberry pi. *Computer*, 46(10):14–16, October 2013.
- [13] Resnick M. et al. Scratch: Programming for All. Communications of the ACM. Št. 11, Str. 60-67
- [14] Redware Research Limited[Online]. Dosegljivo:
<http://scratch.redware.com/content/history-scratch>.