222023112 Max Phillepus

222102055 Elvis Petrus

222072059 Nevell Shigwedha

222064013 Alfeus Kautwima

222119837 Veroraije Tjaapo

221118764 Johnson Amuthenu

221006613 Ngundaa Murorua

SECTION A: Algorithms representation of the different modules and/functions (Pseudocode/flowchart)

pseudo code below;

initialize music player

initialize playlist

initialize tracklist

initialize repeat function

initialize search function

initialize add/remove function

play music player

play tracks from playlist

repeat tracks as necessary

search for specific tracks

add/remove tracks as necessary

exit

end

**code**

algorithm below;

function create\_playlist()

tracks := 100

linked\_list := create\_new\_linked\_list()

for i := 0 to tracks-1 do

add\_track(linked\_list, i)

end for

return linked\_list

end function

function play\_track(linked\_list, track\_number)

current\_track := linked\_list.head

for i := 0 to track\_number-1 do

current\_track := current\_track.next

end for

play\_audio(current\_track.data)

end function

function add\_track(linked\_list, track\_number)

new\_track := create\_new\_track(track\_number)

if linked\_list.head = null then

linked\_list.head := new\_track

else

current\_track := linked\_list.head

while current\_track.next != null do

current\_track := current\_track.next

end while

current\_track.next := new\_track

end if

end function

function remove\_track(linked\_list, track\_number)

current\_track := linked\_list.head

previous\_track := null

for i := 0 to track\_number-1 do

previous\_track := current\_track

current\_track := current\_track.next

end for

if previous\_track = null then

linked\_list.head := current\_track.next

else

previous\_track.next := current\_track.next

end if

end function

function search\_playlist(linked\_list, track\_number)

current\_track := linked\_list.head

while current\_track != null do

if current\_track.data = track\_number then

return true

end if

current\_track := current\_track.next

end while

return false

end function

SECTION B

public class musicplayer {  
 public static void main(String[] args) {  
  
 }  
  
 public final class DoubleClass<Music> {  
  
 private Music data;  
 private DoubleClass<Music> next;  
 private DoubleClass<Music> prev;  
  
 public DoubleClass(final DoubleClass<Music> prev, final Music data, final DoubleClass<Music> next) {  
 this.data = data;  
 this.next = next;  
 this.prev = prev;  
 }  
  
  
 public DoubleClass(final Music data) {  
 this(null, data, null);  
 }  
  
 public Music getData() {  
 return data;  
 }  
  
 public DoubleClass<Music> getNext() {  
 return next;  
 }  
  
  
 public void setTheNextTrack(final DoubleClass<Music> next) {  
 this.next = next;  
 }  
  
 public void setThePreviousTrack(final DoubleClass<Music> prev) {  
 this.prev = prev;  
 }  
  
 public DoubleClass<Music> getPrev() {  
 return prev;  
 }  
  
 public void addTracks(final Music data) {  
 this.data = data;  
 }  
  
 }  
}