TALLINNA TEHNIKAÜLIKOOL

INFOTEHNOLOOGIA TEADUSKOND

Arvutitehnika instituut

Süsteemitarkvara õppetool

 MAHB

IAG0581 Programmeerimine I

**VÕIDUSÕIT**

Kodutöö nr.2

Juhendaja: dotsent Vladiimir Viies

Tallinn 2013

Olen koostanud antud töö iseseisvalt. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

Sisukord:

Ülesande püstitus 4

Algoritm 5

Programmikood 6

Programmi seletus 11

Pilt programmist 12

Kasutatud kirjandus 13

Ülesande püstitus

TEEMA: Võidusõit

On antud kaks massiivi - võidusõidus osalevate autode kaugus finishist ning autode kiirus.

Kasutaja käest küsitakse tundide arvu t. Kirjutada programm, mis leiab, mitu autot on t

tunni pärast finishis. Teepikkust, kiirust ja aega seob valem: s=v\*t.

Lahenduses kasuta funktsioone:

● Funktsioon kauguste ja kiiruste väljastamiseks ekraanile (üks funktsioon, mitu

pöördumist)

● Funktsioon kauguste muutmiseks vastavalt sisestatud tundide arvule – iga auto

kaugust vähendada antud aja jooksul läbitud teepikkuse võrra

(teepikkus=kiirus\*aeg). Kui tulemus tuleb negatiivne, siis asenda see nulliga.

● Funktsioon, mis leiab, mitu autot on juba finishis (e. mitme auto kaugus finishist on

null)

Soovi korral kasuta ka lisafunktsioone. Programmi koostamisel kasuta lihtsuse ja aja

kokkuhoiu mõttes algväärtustatud massiive. Autodega seonduvad maksimaalsed arvud

määra #define direktiiviga või võtmesõna const abil.

Algorütm:



Programmikood:

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define PILOT\_NAMES "F.Alonso", "M.Webber", "T.Glock", "S.Vettel", "J.Button"

#define NAME\_LEN 20

#define COUNT 5

#define MIN\_DIS 100 // min distants

#define MAX\_DIS 300 // max distants

#define MIN\_SPE 60 // min kiirus

#define MAX\_SPE 180 // max kiirus

int genDistance(); //funktsiooni prototyybid

int genSpeed();

int \*genKaugus();

int \*genKiirus();

void PrintS6itjad(char Piloodid[COUNT][NAME\_LEN]);

int AjaSisestus();

void PrintL6petajad(char Piloodid[COUNT][NAME\_LEN]);

int main()

{

 char pilots[COUNT][NAME\_LEN] = {PILOT\_NAMES}; // S6itja nimede jada

 PrintS6itjad(pilots); // Kutsub esile tabeli printimise funktsiooni

 PrintL6petajad(pilots); // Kutsub esile l6pptulemuse

 return 0;

}

int genDistance() // Genereerib Min ja max kaugused

{

 return (rand() % (MAX\_DIS - MIN\_DIS)) + MIN\_DIS;

}

int genSpeed() // Genereerib min ja max kiirused

{

 return (rand() % (MAX\_SPE - MIN\_SPE)) + MIN\_SPE;

}

int \*genKaugus() // Genereerib igale piloodile tema kauguse finisist

{

 int i, \*Kaugus = malloc(sizeof(int) \* COUNT);

 srand(time(NULL));

 i = 0;

 while(i < COUNT)

 {

 Kaugus[i] = genDistance(MIN\_DIS, MAX\_DIS);

 i++;

 }

 return Kaugus; // V2ljastab genereeritud kaugused

}

int \*genKiirus() // Genereerib igale piloodile konstantse kiiruse

{

 int i, \*Kiirus = malloc(sizeof(int) \* COUNT);

 srand(time(NULL));

 i = 0;

 while(i < COUNT)

 {

 Kiirus[i] = genSpeed(MIN\_DIS, MAX\_DIS);

 i++;

 }

 return Kiirus; // V2ljastab genereeritud kiirused

}

void PrintS6itjad(char Piloodid[COUNT][NAME\_LEN]) // Prindib v2lja sõitjate tabeli koos kauguste ja kiirustega

{

 int i, AutodeKaugus[COUNT], AutodeKiirus[COUNT];

 i = 0;

 while(i < COUNT)

 {

 AutodeKaugus[i] = \*(genKaugus() + i); // Kirjutab eelneva kaugust genereeriva funktsiooni v2ljastuse uude jadasse

 AutodeKiirus[i] = \*(genKiirus() + i); // Kirjutab eelneva kiirust genereeriva funktsiooni v2ljastuse uude jadasse

 i++;

 }

 printf("S6itja - Kaugus finishist (km) - Kiirus (km/h):\n");

 i = 0;

 while(i < COUNT) // Selles tsyklis toimub tabeli v2ljaprintimine

 {

 printf("%s", Piloodid[i]);

 printf("%13d", AutodeKaugus[i]);

 printf("%20d\n", AutodeKiirus[i]);

 i++;

 }

}

int AjaSisestus() // Küsib kasutajalt tundide arvu

{

 int t;

 t = 0;

 printf("Sisestage aeg (tundides)\n");

 scanf("%d", &t);

 return t; // V2ljastab aja v22rtuse t2isarvuna

}

void PrintL6petajad(char Piloodid[COUNT][NAME\_LEN]) // Arvutab kasutaja poolt sisestatud tundide arvuga uue l6ppkauguse igale piloodile ja

{ // Prindib v2lja autode l6ppkauguse koos l6petanute arvuga

 int i, t, sum, MasinateKaugus[COUNT], MasinateKiirus[COUNT];

 i = 0;

 while(i < COUNT)

 {

 MasinateKaugus[i] = \*(genKaugus() + i); // Kirjutab eelneva kaugust genereeriva funktsiooni v2ljastuse uude jadasse

 MasinateKiirus[i] = \*(genKiirus() + i); // Kirjutab eelneva kiirust genereeriva funktsiooni v2ljastuse uude jadasse

 i++;

 }

 i = 0;

 t = AjaSisestus(); // V2ljastatud aja v22rtus kirjutatakse uuesti ymber

 sum = 0;

 while(i < COUNT)

 {

 MasinateKaugus[i] = MasinateKaugus[i] - (MasinateKiirus[i]\*t); // Arvutab pilootide l6ppkaugused

 i++;

 }

 printf("Autode uus kaugus finishist:\n");

 i = 0;

 while (i < COUNT)

 {

 printf("%s", Piloodid[i]);

 if(MasinateKaugus[i] > 0)

 {

 printf(" %d\n", MasinateKaugus[i]); // Prindib v2ljam, kui ei j6udnud finishisse

 }

 else

 {

 printf(" 0\n"); // Prindib v2lja k6ik finishit yletanud piloodid

 sum++;

 }

 i++;

 }

 printf("Finishisse j6udnud autode arv on:\n%d\n", sum);

}Programmi seletus:

Programm koosneb kolmest peaosast: parameetrite genereerimisest, aja sisestusest koos lõppkauguse töötlusega ning väljastusest. Esimeses osas genereerib programm igale piloodile tema kauguse finišhist, konstantse kiiruse ja väljastab algandmed tabelina.

Teises osas toimub aja sisestus koos uute kauguste arvutamisega finišhist.

Kolmandas osas väljastatakse kõikide pilootide nimed koos uuete kaugusega finišhist. Kui piloot on jõudnud finišhini, siis väljastatakse tema kaugus kui 0 km. Samuti väljastatakse finišhisse jõudnute pilootide arv.

Pilt programmist:



Kasutatud kirjandus:

<http://stackoverflow.com/questions/8437791/c-malloc-array-in-function-and-then-access-array-from-outside>

<http://stackoverflow.com/questions/7578458/create-a-string-array-with-characters-in-c>

<http://stackoverflow.com/questions/14297169/how-to-make-an-array-return-type-from-c-function>