Лаб№11

1. Відомі дані про ріст 15 юнаків класу. Впорядкувати дані за спаданням. Жодна пара учнів не має однаковий зріст. На початку навчального року в клас з’явився новий учень (відомо, що його ріст не збігається з ростом жодного з учнів класу, не перевищує ріст найнижчого учня та менший росту найвищого).
   1. Вивести прізвища всіх учнів, ріст яких менше росту новенького.
   2. Визначити прізвище учня, після якого варто записати прізвище новенького, щоб упорядкованість не порушилася.
   3. Визначити прізвище учня, ріст якого найменше відрізняється від росту новенького.
   4. Одержати новий список прізвищ учнів (з урахуванням прізвища новенького), у якому прізвища розташовані в порядку спадання росту.

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

typedef struct {

char c[100];

int h;

}g;

void p(g infa){

printf("last name %s \t height %d \n", infa.c,infa.h);};

int main(){

char cc[100],cn[100];

int hh,hn, v=300;

int i,n,j=0,t,x;

g \*inf;

printf("n=");

scanf("%d",&n);

inf = (g \*) calloc (n,sizeof(g));

if(inf == 0){exit(1);}

printf("\n");

for (i=0; i<n; i++){

printf("last name ");

fflush(stdin);

gets(inf[i].c);

printf("height ");

scanf("%d",&inf[i].h);}

printf("last name new one ");

fflush(stdin);

gets(cn);

printf("height ");

scanf("%d",&hn);

printf("massiv------ \n");

while(v>0){

for (i=0; i<n;i++){

t=inf[i].h;

if(t==v){

p(inf[i]); v--;

}

}v--;}

/\* 1 \*/

printf("\n menshe \n");

for (i=0; i<n;i++){t=inf[i].h;

if(t<hn){p(inf[i]);x=i;}}

/\* 1 \*/

/\* 2 \*/

int max=-999;

for (i=0; i<n; i++){

if((inf[i].h<hn)&&(inf[i].h>max)){

max=inf[i].h; x=i;

}

}

printf("\n new after = ");p(inf[x]);

/\* 2 \*/

/\* 3 \*/

int u,min=999;

for (i=0; i<n; i++){

if((inf[i].h>hn)&&(inf[i].h<min)){

min=inf[i].h; u=i;

}

}

max=hn-max;

min=min-hn;

if(max>min){printf("\n naymenshe = ");p(inf[u]);}else{printf("\n naymenshe = ");p(inf[x]);}

/\* 3 \*/

/\* 4 \*/

printf("new------ \n");

v=300;

while(v>0){

if(v==hn){

printf("last name %s \t height %d \n", cn,hn);

}

for (i=0; i<n;i++){

t=inf[i].h;

if(t==v){

p(inf[i]);

}

}v--;}

system("pause");

return 0;

}

ПРОСТІ ЗАДАЧІ

Відомі дані про вартість кожного з 20 найменувань товарів: число гривень та число копійок. Скласти програму, що порівнює вартість двох будь-яких найменувань товарів (визначальну, який з товарів коштує дорожче).

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

typedef struct {

char c[100];

int h,k;

}g;

void p(g infa){

printf("name %s \t grn %d kopiyok %d \n", infa.c,infa.h, infa.k);};

int main(){

int i,n,a,b,hh,kk;

g \*inf;

printf("n=");

scanf("%d",&n);

printf("vvedit nomer a");

scanf("%d",&a);

printf("vvedit nomer b");

scanf("%d",&b);

inf = (g \*) calloc (n,sizeof(g));

if(inf == 0){exit(1);}

printf("\n");

for (i=0; i<n; i++){

printf("name");

fflush(stdin);

gets(inf[i].c);

printf("grn");

scanf("%d",&inf[i].h);

printf("kopiyok");

scanf("%d",&inf[i].k);

}

hh=inf[a].h\*100+inf[a].k;

kk=inf[b].h\*100+inf[b].k;

If(hh>kk{printf(“\n bilshe” ); p(inf[a]); printf(“\n menshe” ); p(inf[b]);

} else {printf(“\n menshe” ); p(inf[a]); printf(“\n bilshe” ); p(inf[b]);

}

system("pause");

return 0;

}