



REPUBLIKA E SHQIPËRISË
MINISTRIA E ARSIMIT
DHE SPORTIT
QENDRA E SHËRBIMEVE ARSIMORE
DREJTORIA E VLERËSIMIT

OLIMPIADA KOMBËTARE E INFORMATIKËS
NË ARSIMIN E MESËM TË LARTË

Klasa 11

Faza e dytë

Viti shkollor: 2023-2024

ZGJIDHJE

Ushtrimi 1. Shkruani një program që merr 5 numra nga përdoruesi dhe gjen sasinë e numrave tek që plotpjesëtohen me 3.

5 pikë

Zgjidhja e ushtrimit 1:

```
#include <stdio.h>
```

```
int main() {
```

```
    // Deklaro variablat
```

```
    int numri[5];
```

```
    int count = 0;
```

```
    // Merr vlerat nga përdoruesi
```

```
    printf("Vendosni pesë numrat:\n");
```

```
    for (int i = 0; i < 5; i++) {
```

```
        printf("numri %d: ", i + 1);
```

```
        scanf("%d", &numri[i]);
```

```
        // Kontrollon nëse numrat janë tek dhe të plotpjesëtueshëm nga 3
```

```
        if (numri[i] % 2 != 0 && numri[i] % 3 == 0) {
```

```
            count++;
```

```

}

}

// Shfaq rezultatet

printf("Numri i numrave tek të plotëpjesëtueshëm nga 3 është: %d\n", count);

return 0;

}

```

Ushtrimi 2. Shkruani një program i cili modifikon një varg të caktuar duke zëvendësuar karakteret e përsëritura me karakterin dhe numrin e përsëritjes së tij. **7 pikë**

Shembull:

Input: aaaaabbbccd

Output: a5b3c2d1

Zgjidhja e ushtrimit 2:

```

#include <stdio.h>
#include <string.h>

void modifikoVargun(char *vargu) {
    int gjatesia = strlen(vargu);

    if (gjatesia == 0) {
        return;
    }

    char rezultati[2 * gjatesia]; // Vargu rezultues mund të jetë maksimumi dy herë më i gjatë se ai origjinal

    int index_rezultati = 0;
    int count = 1;

    for (int i = 0; i < gjatesia; i++) {
        if (vargu[i] == vargu[i + 1]) {
            count++;
        } else {
            rezultati[index_rezultati++] = vargu[i];
            rezultati[index_rezultati++] = count + '0'; // Konverto numrin e përsëritjes në karakter dhe shto në rezultat
            count = 1;
        }
    }
}

```

```
rezultati[index_rezultati] = '\0'; // Shto karakterin e fundit të vargut rezultues
```

```
// Kopjo vargun rezultues në vargun origjinal
strcpy(vargu, rezultati);
}
```

```
int main() {
    char vargu[100];

    printf("Shkruani vargun: ");
    fgets(vargu, sizeof(vargu), stdin);

    // Hiq karakterin nga fundi i vargut nëse ka
    char *newline = strchr(vargu, '\n');
    if (newline != NULL) {
        *newline = '\0';
    }

    modifikoVargun(vargu);

    printf("Vargu i modifikuar: %s\n", vargu);

    return 0;
}
```

Ushtrimi 3. Jepet vektori $N [1:n]$ me emrat e n nxënësve dhe vektori $A [1:n]$ me numrin e mungesave të secilit prej tyre në lëndën e informatikës. Të ndërtohet programi i cili heq nga vektorët nxënësin me pjesëmarrjen më të ulët, së bashku me numrin e mungesave të tij.

12 pikë

Zgjidhja e ushtrimit 3:

```
#include <stdio.h>
#include <string.h>

int main() {
    int n;

    // Merr numrin e nxënësve
    printf("Shkruani numrin e nxënësve: ");
    scanf("%d", &n);

    // Deklaro vektorët e emrave dhe numrit të mungesave
    char emrat[n][50];
    int mungesat[n];

    // Mbush vektorët me të dhëna
```

```
for (int i = 0; i < n; i++) {
    printf("Emri i nxënësit %d: ", i + 1);
    scanf("%s", emrat[i]);
    printf("Numri i mungesave për %s: ", emrat[i]);
    scanf("%d", &mungesat[i]);
}

// Gjej nxënësin me numrin më të madh të mungesave
int max_mungesat = mungesat[0];
int index_max_mungesat = 0;

for (int i = 1; i < n; i++) {
    if (mungesat[i] > max_mungesat) {
        max_mungesat = mungesat[i];
        index_max_mungesat = i;
    }
}

// Printo informacionin për nxënësin me numrin më të madh të mungesave
printf("\nNxënësi me numrin më të madh të mungesave:\n");
printf("Emri: %s\n", emrat[index_max_mungesat]);
printf("Numri i mungesave: %d\n", max_mungesat);

// Heq nxënësin nga vektorët
for (int i = index_max_mungesat; i < n - 1; i++) {
    strcpy(emrat[i], emrat[i + 1]);
    mungesat[i] = mungesat[i + 1];
}

n--; // Zvogëlo numrin e nxënësve pas heqjes

// Printo vektorin e përditësuar
printf("\nVektori i përditësuar:\n");
for (int i = 0; i < n; i++) {
    printf("Emri: %s, Mungesat: %d\n", emrat[i], mungesat[i]);
}

return 0;
}
```

Ushtrimi 4. Jepet n , një numër i plotë pozitiv. Shkruani një program i cili gjeneron një matricë katrore të madhësisë $n \times n$, në mënyrë që diagonalja kryesore të përbëhet nga numra të thjeshtë, të ndryshëm nga njëri tjetri. (Numër i thjeshtë quhet numri i cili plotpjesëtohet vetëm me numrin 1 dhe veten e tij.) **13 pikë**

Zgjidhja e ushtrimit 4:

```
#include <stdio.h>

// Funkcioni për të kontrolluar nëse një numër është i thjeshtë
int isPrime(int num) {
    if (num < 2) {
        return 0; // Numri nuk është i thjeshtë
    }
    for (int i = 2; i * i <= num; i++) {
        if (num % i == 0) {
            return 0; // Numri nuk është i thjeshtë
        }
    }
    return 1; // Numri është i thjeshtë
}

int main() {
    int n;
    // Leximi i madhësisë së matricës nga përdoruesi
    printf("Shkruani madhësinë e matricës (n x n): ");
    scanf("%d", &n);
    // Krijimi i matricës
    int matrix[100][100];

    // Gjenerimi i diagonalitës kryesore me numra të thjeshtë
    int primeNum = 2;
    for (int i = 0; i < n; i++) {
```

```

matrix[i][i] = primeNum;

// Gjej numrin tjetër të thjeshtë për diagonalën kryesore
do {
    primeNum++;
} while (!isPrime(primeNum));
}

```

// Gjenerimi i elementeve të tjera të matricës

```

int num = 1;
for (int i = 0; i < n; i++) {
    for (int j = 0; j < n; j++) {
        if (matrix[i][j] == 0) {
            matrix[i][j] = num++;
        }
    }
}

```

// Printimi i matricës

```

printf("\nMatrica e krijuar:\n");
for (int i = 0; i < n; i++) {
    for (int j = 0; j < n; j++) {
        printf("%d ", matrix[i][j]);
    }
    printf("\n");
}

return 0;

```

Ushtrimi 5. Të ndërtohet programi i cili analizon produktivitetin e punonjësve në një kompani për një javë. Përdoruesit duhet të kenë mundësi të shtojnë punonjës të rinj, të shkruajnë detyrat e tyre të përditshme të përfunduara dhe orët e punës. Në përfundim duhet të shfaqet totali i detyrave të përfunduara, mesatarja e numrit të detyrave të kryera në ditë dhe punonjësi më produktiv i javës.

13 pikë

Zgjidhja e ushtrimit 5:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAX_PUNONJES 50
#define MAX_DETYRA 100
struct Punonjes {
    char emri[50];
    int ore_pune;
    int detyrat_e_kryera;
};
struct Punonjes punonjesit[MAX_PUNONJES];
int numri_punonjesve = 0;
void shtoPunonjes() {
    if (numri_punonjesve < MAX_PUNONJES) {
        printf("Emri i punonjesit: ");
        scanf("%s", punonjesit[numri_punonjesve].emri);
        printf("Oret e punes per javen e kaluar: ");
        scanf("%d", &punonjesit[numri_punonjesve].ore_pune);
        printf("Numri i detyrave te kryera: ");
        scanf("%d", &punonjesit[numri_punonjesve].detyrat_e_kryera);
        numri_punonjesve++;
    } else {
        printf("Nuk mund te shtohen me shume punonjes.\n");
    }
}
void shfaqRaportin() {
    int total_detyra = 0;
    float mesatarja_detyrave = 0;
```

```
int max_detyra = 0;

char emri_max_punonjesi[50];

for (int i = 0; i < numri_punonjesve; i++) {

    total_detyra += punonjesit[i].detyrat_e_kryera;

    mesatarja_detyrave += punonjesit[i].detyrat_e_kryera;

    if (punonjesit[i].detyrat_e_kryera > max_detyra) {

        max_detyra = punonjesit[i].detyrat_e_kryera;

        strcpy(emri_max_punonjesi, punonjesit[i].emri);

    }

}

if (numri_punonjesve > 0) {

    mesatarja_detyrave /= numri_punonjesve;

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

    printf("Totali i detyrave te pefunduara: %d\n", total_detyra);

    printf("Mesatarja e detyrave te kryera ne dite: %.2f\n", mesatarja_detyrave);

    printf("Punonjesi më produktiv: %s\n", emri_max_punonjesi);

} else {

    printf("Nuk ka te dhena per te shfaqur.\n");

}

}

int main() {

    int opsioni;

    do {

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

        printf("1. Shto punonjes\n");

        printf("2. Shfaq raportin\n");

        printf("0. Dil\n");

        printf("Zgjedhja juaj: ");
```



```
scanf("%d", &opsioni);  
switch (opsioni) {  
    case 1:  
        shtoPunonjes();  
        break;  
    case 2:  
        shfaqRaportin();  
        break;  
    case 0:  
        printf("Dilim nga programi.\n");  
        break;  
    default:  
        printf("Opsioni e padukshme.\n");  
}  
} while (opsioni != 0);  
return 0;  
}
```