

prvi test

1

```
int a = 0xAB;
int b = 036;
int c = a & b;
int d = (a | b ) ^ c;
printf("%d %x %o\n", c, c, c);
printf("%d %x %o\n", d, d, d);
```

odgovor:

```
10 a 12
181 b5 265
```

2

```
int a = 0126;
int b = 0x14;
int c = a << 2;
int d = (a ^ b) & c;
printf("%d %x %o\n", c, c, c);
printf("%d %x %o\n", d, d, d);
```

odgovor:

```
344 158 530
64 40 100
```

3

```
int a = 0xFF;
int b = 067;
int c = (a << 3) | b;
int d = c ^ b;
printf("%d %d %d %d", a, b, c, d);
```

odgovor:

```
255 55 2047 1992
```

4

```
int a = 0xAB;
int b = 054;
int c = b | (a >> 2);
int d = c ^ a;
printf("%d %d %d %d", a, b, c, d);
```

odgovor:

171 44 46 133

5

```
printf("%c %o %x", '\n', '\n', '\n');
```

odgovor:

\n 12 a

6

```
printf("printf(\\n\\t//n//t)");
```

odgovor:

```
printf(\
\\t//n//t)
```

7

```
printf("//n//t\\n\\t");
```

odgovor:

//n//t\\n\\t

8

```
double a = 0.6;
printf("%.2f %lu", a, sizeof(a));
```

odgovor:

0.60 8

9

```
char a = 'e';
printf("%d %lu", a, sizeof(a));
```

odgovor:

101 1

10

```
printf("%c", rand() % 25 + 65);
```

odgovor:

65..89

11

```
printf("%c", rand() % 25 + 97);
```

odgovor:

97..121

12

```
printf("%c %o %x", '\\', '\\', '\\');
```

odgovor:

\ 134 5c

13

```
printf("%c %x", 'z' - 32, 'z' - 32);
```

odgovor:

Z 5a

14

```
printf("\\\\");
```

odgovor:

\"

15

```
printf("%d + 3", 1 && 0);
```

odgovor:

0 + 3

16

```
printf("%d", '5' - '0');
```

odgovor:

5

17

```
putchar(50);
```

odgovor:

2

18

```
putchar(0x4A);
```

odgovor:

J

19

```
printf("%c %x", 'C' + 32, 'C' + 32);
```

odgovor:

C 63

20

```
printf("\\\\\\\\n");
```

odgovor:

\\\\n

21

```
printf("%d - 3", '2' - '0');
```

odgovor:

2 - 3

22

```
printf("%d", 0xFF);
```

odgovor:

255

23

```
printf("%d", rand() % 5 - 10);
```

odgovor:

-10, -9, -8, -7, -6

24

```
printf("%lu", sizeof(double) - sizeof(float));
```

odgovor:

4

25

```
int a = 11, b=3;
float c = 2.0;
a %= b++;
float d = --b / c;
printf("%d %d %f", a, b, c);
```

odgovor:

2 3 2.000000

26

```
int a = 11, b=3;
float c = 2.0;
c /= b++;
a /= b;
printf("%d %d %f", a, b, c);
```

odgovor:

2 4 0.666667

27

```
int a = 12; a /= 3;
float b = --a / 2;
printf("%d %f", a, b);
```

odgovor:

3 1.000000

28

```
int i = 1, j = 0, k = 1, m;
m = !(i && j++) || ++k;
printf("%d %d %d %d", i, j, k, m);
```

odgovor:

1 1 1 1

29

```
int i = 1, j = 0, k = 1, m;
m = (i++ || ++j) && k--;
printf("%d %d %d %d", i, j, k, m);
```

odgovor:

2 0 0 1

30

```
int i = 0, j = -1, k = 1, m;  
m = !(i++ && ++j) || ++k;  
printf("%d %d %d %d", i, j, k, m);
```

odgovor:

1 -1 1 1

31

```
int i = 0, j = 0, k = 1, m;  
m = (++i || ++j) && !(i - k--);  
printf("%d %d %d %d", i, j, k, m);
```

odgovor:

1 0 0 1

32

```
int i = 0, j = 1, k = 1, m;  
m = (++i && (j - 1)) || (--k + j++ );  
printf("%d %d %d %d", i, j, k, m);
```

odgovor:

1 2 0 1

33

```
int i = 2;  
int a = ++i > 2 ? i % 2 == 0 ? i-- : i++ : i + 3;  
printf("%d %d", a, i);
```

odgovor:

3 4

34

```
int i = 3;  
int a = --i > 2 ? i == 1 ? i + 1 : i++ : i--;  
printf("%d %d", a, i);
```

odgovor:

2 1

35

```
int a = 9, b = 2;
b++;
a %= b + 1;
float c = ++a / 4;
printf("%d %d %.1f", a, b, c);
```

odgovor:

2 3 0.0

36

```
int a = 8, b = 3;
b++;
a %= b + 1;
float c = a++ / 2;
printf("%d %d %.1f", a, b, c);
```

odgovor:

4 4 1.0

37

```
int a = 8, b = 47, c = ++b / a--, d = b++ * --a;
printf("%d %d %d %d", a, b, c, d);
```

odgovor:

6 49 6 288

38

```
int a, b, c, d;
a = b = -6;
c = 210 / a;
d = 210U / b;
printf("%d %d %d %d", a, b, c, d);
```

odgovor:

-6 -6 -35 0

39

```
int p, q, r, s, t;
p = 0x5BC7;
q = 0xB55E;
r = p & q;
s = p | q;
t = p ^ q;
printf("%d %d %d %d %d", p, q, r, s, t);
```

odgovor:

23495 46430 4422 65503 61081

40

```
int a, b, c;
double x, y;
a = 31; b = 6; c = 64;
x = a++ / --b;
y = ++c / b--;
printf("%d %d %d %f %f", a, b, c, x, y);
```

odgovor:

32 4 65 6.000000 13.000000

41

```
int a = 26, b = 6, c;
c = (--a > ++b) ? --a * b : a / ++b;
printf("%d %d %d", a, b, c);
```

odgovor:

24 7 168

42

```
int a = -19, b = 42, c;
c = (a++ < 115U) ? ++b : b--;
printf("%d %d %d", a, b, c);
```

odgovor:

-18 41 42

43

```
int a = 9, b = 16, c;  
c = (a = b) ? a++ * --b : --a / b++;  
printf("%d %d %d", a, b, c);
```

odgovor:

17 15 240

44

```
int x, y, z, t;  
x = 13;  
y = 11;  
z = 17;  
t = (x > y || ++y == --z) ? x * y * x : y * x;  
printf("%d %d %d %d", x, y, z, t);
```

odgovor:

13 11 17 1859

45

```
int u, v, w;  
u = 0x659C;  
v = 27;  
w = (u & 1 << 8) ? v++ << 1 : --v >> 2;  
printf("%d %d %d", u, v, w);
```

odgovor:

26012 28 54

46

```
int k, m, n;  
m = 72;  
n = 23;  
k = (m >>= 2) ? --n << 1 : n++ >> 2;  
printf("%d %d %d", k, m, n);
```

odgovor:

44 18 22

47

```
int a = 2;
int b = (a-- && (a == 1)) ? a-- : a + 3;
printf("%d %d", a, b);
```

odgovor:

0 1

48

```
int a = 1;
int b = (--a && (a == 1)) ? a-- : ++a;
printf("%d %d", a, b);
```

odgovor:

1 1

49

```
int i = 7, j = 1;
do {
    j++;
    i = i - 2;
    if(i < 0 || i % j == 0)
        continue;
    printf("iteracija\n");
}while(i > 0);
printf("%d %d", i, j);
```

odgovor:

iteracija

iteracija

-1 5

50

```
int i = 6, j = 2;
do {
    i -= 2;
    if(i % 2 == 0 && j % 2 == 0)
        continue;
    j++;
    printf("iteracija\n");
}while(i);
printf("%d %d", i, j);
odgovor:
0 2
```

51

```
int i = 4, j = 2;
do {
    i--;
    if(i % 2 == 0 && j % 2 == 0) {
        j++;
        continue;
    }
    printf("iteracija\n");
}while(i);
printf("%d %d", i, j);
odgovor:
iteracija
iteracija
iteracija
0 3
```

52

```
int i = -3, j = 0;
while(i < 0 || j < 1) {
    i++;
    if(i == -2)
        continue;
    j++;
    printf("iteracija\n");
}
printf("%d %d", i, j);
```

odgovor:

```
iteracija
iteracija
0 2
```

53

```
int i, j;
for(i = 2; i < 7; i++) {
    for(j = 2; j < i; j++) {
        if(i % j == 0)
            break;
    }
    printf("%d ", j);
}
printf("\nposle %d", i);
```

odgovor:

```
2 3 2 5 2
posle 7
```

54

```
int i;
for(i = 436; i > 0; i /= 10) {
    if(i % 10 == 6)
        continue;
    printf("iteracija\n");
    int j = i % 10;
    int k = 1;
    while(j) {
        j--;
        k++;
        if(j==k)
            break;
        j--;
    }
    printf("%d %d\n",j,i);
}
```

odgovor:

```
iteracija
2 43
iteracija
0 4
```

55

```
int i;
for(i = 432; i > 0; i /= 10) {
    if(i % 10 == 3)
        continue;
    printf("iteracija\n");
    int j = i % 10;
    int k = 0;
    while(j) {
        j--;
        k++;
        if(j==k)
            break;
        j--;
    }
    printf("%d %d\n",j,i);
}
```

odgovor:

```
iteracija
1 432
iteracija
0 4
```

56

```
int i = 0, j;  
do {  
    i++;  
    if(i % 2 != 0)  
        continue;  
    printf("%d ", i);  
    j = i;  
    while( j > 0) {  
        j--;  
        if(j % 2 == 0)  
            break;  
        printf("%d\n", j);  
    }  
}while(i < 6);
```

odgovor:

```
2 1  
4 3  
6 5
```

57

```
void f(int b) {
    b += 4;
}
int main() {
    int b = 2;
    f(b);
    printf("%d", b);
    return 0;
}
```

odgovor:

2

58

```
int x = 1;
int f(int x, int a) {
    a += x;
    return a;
}
int main() {
    int rez = f(2, x);
    printf("%d %d", rez, x);
    return 0;
}
```

odgovor:

3 1

59

```
int f(int x, int a) {
    a += x;
    return a;
}
int main() {
    int rez = f(x + 2, 3);
    printf("%d %d", rez, x);
    return 0;
}
```

odgovor:

compile error: use of undeclared identifier 'x'

60

```
int main() {  
    int a = 0;  
    while(a < 3) {  
        int b = a;  
        b++;  
        a++;  
    }  
    printf("%d", a + b);  
}
```

odgovor:

compile error: use of undeclared identifier 'b'

drugi test

1

```
int main() {
    int i = 0;
    char c = 'a';
    while(i < 2) {
        i++;
        switch(c) {
            case 'a':
                printf("a");
                break;
        }
    }
    printf("\nposle");
    return 0;
}
```

odgovor:

aa

posle

2

```
int main() {
    int i = 0, a = 0;
    for(; i < 5; i++) {
        a++;
        if(i == 3)
            break;
    }
    printf("%d", a);
    return 0;
}
```

odgovor:

4

3

```
int main() {  
    int i = 0;  
    for(i = 0; i < 5; i++)  
        if(i < 4) {  
            printf("iteracija\n");  
            break;  
        }  
    return 0;  
}
```

odgovor:

iteracija

4

```
int main() {  
    int i = 0, j = 0;  
    for(i = 0; i < 5; i++) {  
        for(j = 0; j < 4; j++) {  
            if(i > 1)  
                continue;  
            printf("iteracija\n");  
        }  
    }  
    return 0;  
}
```

odgovor:

iteracija
iteracija
iteracija
iteracija
iteracija
iteracija
iteracija
iteracija

5

Napisati funkciju `int max_index(double a[], int n)` koja određuje indeks najvećeg elementa u nizu `a` veličine `n`.

odgovor:

```
int max_index(double a[], int n) {
    if(!a || n < 1)
        return -1;
    int i, index = 0, max = a[index];
    for(i = 0; i < n; i++)
        if(a[i] > max)
            max = a[index = i];
    return index;
}
```

6

Napisati funkciju `int get_greater(double a[], double b[], double c, int n)` koja iz niza `a` veličine `n` prepisuje u niz `b` elemente veće od `c`. Funkcija vraća broj prepisanih elemenata.

odgovor:

```
int get_greater(double a[], double b[], double c, int n) {
    if(!a || !b || n < 1)
        return -1;
    int i, j = 0;
    for(i = 0; i < n; i++)
        if(a[i] > c)
            b[j++] = a[i];
    return j;
}
```

7

```
int main() {
    int a[4] = {1, 2, 3, 4};
    int* p;
    p = a + 3;
    *p = *a + 5;
    printf("%d %d %d %d", a[0], a[1], a[2], a[3]);

    return 0;
}
```

odgovor:

1 2 3 6

8

```
void f(int* p) {
    p++;
    *p = 6;
}

int main() {
    int a = 1;
    f(&a);
    printf("%d", a);
    return 0;
}
```

odgovor:

1

9

```
int main() {  
    int x = 1;  
    int* p = &x;  
    *p = 5;  
    p++;  
    printf("%d", x);  
    return 0;  
}
```

odgovor:

5

10

```
int main() {  
    int x = 3;  
    int* p = &x;  
    p++;  
    *p = 1;  
    printf("%d", x);  
    return 0;  
}
```

odgovor:

3

11

```
int main() {  
    int a[3] = {1, 1, 1};  
    int* p = a + 2;  
    p--;  
    *p = *a + 2;  
    printf("%d %d %d", a[0], a[1], a[2]);  
    return 0;  
}
```

odgovor

1 3 1

12

```
int main() {  
    int a[3] = {3, 2, 1};  
    int* p = a;  
    p++;  
    *p = *a + 4;  
    printf("%d %d %d", a[0], a[1], a[2]);  
    return 0;  
}
```

odgovor:

3 7 1

13

```
int main() {  
    char s[] = "nova";  
    char *p = s + 2;  
    p++;  
    *p = 'e';  
    printf("%s %s", s, p);  
    return 0;  
}
```

odgovor:

nova e

14

```
int main() {  
    char s[] = "raf";  
    char *p = s + 1;  
    *(p + 1) = 's';  
    printf("%s %s", s, p);  
    return 0;  
}
```

odgovor:

ras as

15

```
int main() {
    char s[] = "a";
    strcpy(s, "b");
    if(!strcmp(s, "ab"))
        strcpy(s, "a");
    strcat(s, "ee");
    printf("%s", s);
    return 0;
}
```

odgovor:

bee

16

```
int main() {
    char s[10] = "a";
    strcpy(s, "b");
    if(!strcmp(s, "ab"))
        strcpy(s, "a");
    strcat(s, "ee");
    printf("%s", s);
    return 0;
}
```

odgovor:

bee

17

```
int main() {
    char s[10] = "aa";
    int i = 0;
    for(i = 0; i < strlen(s); i++) {
        strcat(s, "b");
        printf("%s\n", s);
        strcpy(s, "a");
        printf("%s\n", s);
    }
    printf("%d", i);
    return 0;
}
```

odgovor:

aab
a
1

18

```
int main() {
    char s[] = "aba-ac-a";
    char* t = strtok(s, "-");
    if(strlen(t) == 2)
        strcpy(t, "1");
    printf("%s", t);
    return 0;
}
```

odgovor:

aba

19

```
int main() {
    char s[] = "aba-ac-a";
    char* t = strtok(s, "-");
    t = strtok(NULL, "-");
    if(strlen(t) == 2)
        strcat(t, "a");
    printf("%s", t);
    return 0;
}
```

odgovor:

aca

20

```
char s[10] = "d";
while(strcmp(s, "b") > 0) {
    if(strlen(s) < 3)
        strcat(s, "d");
    else
        strcpy(s, "a");
    puts(s);
}
```

odgovor:

dd

ddd

a

21

```
char s[10] = "a";
while(strcmp(s, "c") < 0) {
    if(strlen(s) < 3)
        strcat(s, "a");
    else
        strcpy(s, "d");
    puts(s);
}
```

odgovor:

```
aa
aaa
d
```

22

```
int main() {
    char s[] = "aba";
    char s1[10];
    strcpy(s1, s);
    strcat(s1, "aba");
    while(strlen(s1) > strlen(s)) {
        if(!strcmp(s1, "aba")) {
            strcpy(s, "a");
        }else {
            s1[strlen(s1) - 1] = '\0';
        }
        puts(s1);
    }
    return 0;
}
```

odgovor:

```
abaab
abaa
aba
```

23

Napisati program koji otvara datoteku i umesto postojećeg sadržaja ispisuje 5 nasumično generisanih brojeva u intervalu 1-10 i njihove korene.

odgovor:

```
int main() {
    FILE* fp = fopen("test.txt", "w");
    srand(time(0));
    if(fp) {
        int i;
        for(i = 0; i < 5; i++) {
            int a = rand() % 10 + 1;
            fprintf(fp, "%d %0.2f\n", a, sqrt(a));
        }
        fclose(fp);
    }
    return 0;
}
```

24

Napisati program koji broji i ispisuje koliko ima cifara u datoteci.

odgovor:

```
int main() {
    FILE* fp = fopen("test.txt", "r");
    if(fp) {
        char c;
        int count;
        while((c = getc(fp)) != EOF) {
            if(c >= '0' || c <= '9') {
                count++;
            }
        }
        printf("%d", count);
        fclose(fp);
    }
    return 0;
}
```

25

Napisati program koji broji koliko u fajlu ima linija i ispisuje broj linija na sistemski izlaz.

odgovor:

```
int main() {
    FILE* fp = fopen("test.txt", "r");
    if(fp) {
        int count = 0;
        char c;
        while((c = getc(fp)) != EOF) {
            if(c == '\n') {
                count++;
            }
        }
        printf("%d", count);
    }
    return 0;
}
```

26

Napisati program koji broji koliko reči ima u fajlu i ispisuje broj reči na sistemski izlaz.

odgovor:

```
int main() {
    FILE* fp = fopen("test.txt", "r");
    if(fp) {
        int count = 0;
        char curr, prev;
        while((curr = getc(fp)) != EOF) {
            if((prev >= 'A' && prev <= 'Z')
                || (prev >= 'a' && prev <= 'z')) {
                count += curr == ' ' || curr == '\n';
            }
            prev = curr;
        }
        printf("%d", count);
    }
    return 0;
}
```

27

Napisati program koji iz datoteke čita 5 linija, gde se u svakoj liniji nalazi jedan ceo broj i njegov koren i smešta ih u dva niza, jedan niz sadrži cele brojeve, a drugi njihove korene u istom redosledu.

odgovor:

```
int main() {
    FILE* fp = fopen("test.txt", "r");
    if(fp) {
        int a[5];
        float root[5];
        int i = 0;
        while(!feof(fp) && i < 5) {
            fscanf(fp, "%d %f", &a[i], &root[i]);
            i++;
        }
    }
    return 0;
}
```

28

Napisati primer dinamičke alokacije niza celih brojeva.

odgovor:

```
int* a = (int*) malloc(n * sizeof(int));
```

29

Napisati primer dinamičkog povećanja veličine niza celih brojeva.

odgovor:

```
a = realloc(a, n * sizeof(int));
```

30

Objasniti osnovne razlike između funkcija malloc i calloc za dinamičku alokacije memorije.

odgovor:

malloc ima jedan argument (broj bajtova koji se alocira), a calloc ima dva argumenta (broj promenljivih koji se alocira i veličinu promenljive) i malloc ne inicijalizuje alocirani prostor, dok ga calloc inicijalizuje.

31

Koja funkcija se koristi za oslobađanje memorijskog prostora?

odgovor:

free

32

Napisati funkciju koja briše sve elemente neuređene liste brojeva koji su deljivi sa parametrom n. Data je struktura kojom je implementirana lista.

```
typedef struct Element {  
    int podatak;  
    struct Element* sledeci;  
} Element;
```

odgovor:

```
Element* remove_mod(Element* root, int n) {  
    Element* tmp = root;  
    Element* prev = NULL;  
    while(tmp) {  
        if(tmp->podatak % n == 0) {  
            if(prev) {  
                prev->sledeci = tmp->sledeci;  
                free(tmp);  
                tmp = prev->sledeci;  
            }else {  
                root = tmp->sledeci;  
                free(tmp);  
                tmp = root;  
            }  
        }else {  
            tmp = (prev = tmp)->sledeci;  
        }  
    }  
    return root;  
}
```

33

Napisati kod kojim se kreira struktura kojom se može implementirati jednostruko povezna lista filmova sa podacima o nazivu, žanru i godini proizvodnje filma i napraviti prvo praznu listu, a onda dodati jedan film.

odgovor:

```
typedef struct Film {
    char naziv[200];
    char zanr[30];
    int godina;
    struct Film* sledeci;
} Film;

int main() {
    Film* root = NULL;
    Film* tmp = (Film*) malloc(sizeof(Film));
    strcpy(tmp->naziv, "Kum");
    strcpy(tmp->zanr, "drama");
    tmp->godina = 1972;
    tmp->sledeci = NULL;
    root = tmp;
    return 0;
}
```


34

Napisati funkciju main koja poziva funkciju f i prosleđuje tri promenljive i objasniti kako se menja (ili ne menja) vrednost promenljivih posle poziva funkcije?

```
void f(int* a, float* b, float c) {  
    *b = c;  
    (*a)--;  
    c = 1.0;  
}
```

odgovor:

```
int main() {  
    int a = 1;  
    float b = 2, c = 3;  
    f(&a, &b, c);  
    return 0;  
}
```

vrednost promenljive a se umanjuje za jedan
vrednost promenljive b se postavlja na vrednost promenljive c
vrednost promenljive c se ne menja

35

Napisati funkciju main koja poziva funkciju f i prosleđuje tri promenljive i objasniti kako se menja (ili ne menja) vrednost promenljivih posle poziva funkcije?

```
void f(int a, int* b, float* c) {  
    *b = 5;  
    a--;  
    *c = 1.0;  
}
```

odgovor:

```
int main() {  
    int a = 1, b = 2;  
    float c = 3;  
    f(a, &b, &c);  
    return 0;  
}
```

vrednost promenljive a se ne menja
vrednost promenljive b se postavlja na 5
vrednost promenljive c se postavlja na 1.0

36

Dat je niz celih brojeva, napisati dva načina da se ispiše memorijska adresa i vrednost trećeg elementa ovog niza.

```
int a[3] = {3, 2, 1};
```

odgovor:

```
adresa prvi način: &a[2]
adresa drugi način: a + 2
vrednost prvi način: a[2]
vrednost drugi način: *(a + 2)
```

37

Dat je string, napisati dva načina da se ispiše memorijska adresa i vrednost četvrtog karaktera ovog stringa.

```
char s[] = "programiranje";
```

odgovor:

```
adresa prvi način: &s[3]
adresa drugi način: s + 3
vrednost prvi način: s[3]
vrednost drugi način: *(s + 3)
```

38

```
void fniz(int a[]) {
    int b = a[0];
    int* p = a + 2;
    a[0] = *p;
    *p = b;
}

int main() {
    int a[3] = {3, 2, 1};
    fniz(a);
    printf("%d %d %d", a[0], a[1], a[2]);
    return 0;
}
```

odgovor

```
1 2 3
```

39

```
void fstr(char s[]) {
    char a = s[0];
    char *p = s + 2;
    s[0] = *p;
    *p = a;
}

int main() {
    char s[] = "raf";
    fstr(s);
    printf("%s", s);
    return 0;
}
```

odgovor:

far