

VARIANTE 2007

7. Pentru definiția alăturată a subprogramului `ex`, stabiliți ce se afișează la apelul `ex(120)`?

```
void ex(int x)
{if(x!=0) {
    printf("%d",x%10); / cout<<x%10;
    ex(x/10);
}
}
```

- a. 021 b. 012 c. 120 d. 21

8. Pentru definiția alăturată a subprogramului `ex`, stabiliți ce se afișează la apelul `ex(120)`?

```
void ex(int x)
{    printf("%d",x%10); /cout<<x%10;
    if(x!=0) ex(x/10);
}
```

- a. 0120 b. 021 c. 210 d. 0210

3. Ce valoare va returna `f(23951)`, pentru funcția `f` definită alăturată?

```
int f(int n){
    if (n==0) return 0;
    else
    {int c= f(n/10);
    if (n%10>c)
        return n%10;
    else return c;
    }}
```

- a. 2 b. 3 c. 5 d. 9

6. Pentru funcțiile `f1` și `f2` definite alăturat, stabiliți care este rezultatul returnat la apelul `f2(305)`?

```
long f1(int c)
{ if (c<1) return 1;
  else return c*f1(c-1);
}

long f2(long n)
{ if (n==0) return 0;
  else return f1(n%10)+f2(n/10);
}
```

- a. 126 b. 3 c. 127 d. 8

6. Pentru funcțiile `f1` și `f2` definite mai jos, stabiliți care este rezultatul returnat la apelul `f1(6)`?

```
long f2(long x){ if (x%2==0) return f2(x/2); else return x; }
```

```
long f1(int c) { if (c<1) return 1; else return f2(c*f1(c-1)); }
```

- a. 720 b. 16 c. 45 d. 360

8. Care este rezultatul tipărit la apelul `f("examenul",0)`?

```
void f(char s[256],int i)
{if (i<strlen(s))
    {f(s,i+1);
    if (strchr("aeiou",s[i])!=0)
        cout<<s[i]; | printf("%c",s[i]);
    }
}
```

- a. eaeu b. examenul c. ueae d. e

3. Se consideră subprogramul `F`, definit alăturat. Care este valoarea expresiei `F(11)`?

```
int F(int x){
    if(x%2==0) return x/2;
    else return F(F(x-1));}
```

- a. 2 b. 5 c. 1 d. 4

7. Pentru definiția subprogramului alăturat stabiliți ce se afișează la apelul $f(5, 1)$.

a. 12345

b. 01234

```
void f(int n, int k)
```

```
{if (k<=n) {  
    printf("%d",n-k); | cout<<n-k;  
    f(n,k+1);  
}}
```

c. 43210

d. 54321

6. Se consideră funcția alăturată $f1$. În urma apelului $f1(7)$, se va afișa:

a. 89

b. 821

c. 8102721

d. 810

```
void f1(int i)
```

```
{  
    if (i<=9)  
    { cout<<i+1; / printf("%d",i+1)  
      f1(i+2);  
      cout<<3*i; / printf("%d",3*i)  
    }  
}
```

2. Ce se va afișa în urma apelului $f1(13)$?

a. 125815

b. 122414

```
void f1(int k)
```

```
{ if (k>3)  
  {cout<<k-1; /printf("%d",k-1);  
   f1(k/2);  
   cout<<k+2; /printf("%d",k+2);}}
```

c. 1268

d. 1255

5. Considerând funcția recursivă alăturată, precizați ce se va afișa pe ecran în urma apelului $f(2)$?

a. 2011

b. 1010

```
void f(int a)
```

```
{int i;  
 printf("%d",a); | cout<<a;  
 for(i=0;i<a;i++) f(i);}
```

c. 2100

d. 2010

4. Fie subprogramul f definit alăturat. În urma executării secvenței $\text{int } x=1; f(x);$ se vor afișa valorile:

a. 432

b. 234

```
void f(int & b)
```

```
{b=b+1;if (b<4) f(b);  
 printf("%d",b); | cout<<b;  
}
```

c. 111

d. 444

5. Pentru subprogramul alăturat, apelul $\text{func}(4)$ are ca rezultat:

a. -1

b. -5

```
int func(int x) {
```

```
if(x<=0) return -1;  
else return 1/func(x-1)-2*func(x-2);
```

c. -6

d. 0

4. Care dintre următoarele subprograme afișează în ordine inversă (începând cu cifra unităților) cifrele unui număr natural, primit ca parametru?

a. `void numar(int a)`

```
{if(a>9)numar(a%10);  
cout<<a/10; | printf("%d",a/10);}
```

b. `void numar(int a)`

```
{cout<<a%10; | printf("%d",a%10);  
if(a>9)numar(a/10);}
```

c. `void numar(int a)`

```
{cout<<a/10; | printf("%d",a/10);  
if(a>9)numar(a%10);}
```

d. `void numar(int a)`

```
{if(a>9)numar(a/10);  
cout<<a%10; | printf("%d",a%10);}
```

5. Fie funcția numara prezentată mai jos:

```
int numara(int x,int y)
{if (y==0) return 0;
  else if (x%y==0) return numara(x,y-1)+1;
  else return numara(x,y-1);}
```

Care este apelul corect al funcției numara pentru a verifica dacă un număr natural n este prim?

- a. if (numara(n,n)==2)
cout<<"prim"; | printf("prim");
- b. if (numara(2,n)==2)
cout<<"prim"; | printf("prim");
- c. if (numara(n,2)==0)
cout<<"prim"; | printf("prim");
- d. if (numara(n,n/2)==2)
cout<<"prim"; | printf("prim");

2. Fie subprogramul cu definiția alăturată. Care dintre următoarele instrucțiuni determină afișarea valorii -7?

```
int f (int n)
{
  if(n<=0) return -1;
  else
    return f(n-1)-2*f(n-2)+3*f(n-3);
}
```

- a. printf("%d",f(3)) | cout<<f(3)
- c. printf("%d",f(5)) | cout<<f(5)

- b. printf("%d",f(6)) | cout<<f(6)
- d. printf("%d",f(4)) | cout<<f(4)

1. Fie subprogramul cu definiția alăturată. Care dintre următoarele instrucțiuni determină afișarea valorii -9?

```
int f (int n)
{
  if(n<=0) return -1;
  else
    return f(n-1)-2*f(n-2)+3*f(n-3);
}
```

- a. printf("%d",f(4)) | cout<<f(4)
- c. printf("%d",f(5)) | cout<<f(5)

- b. printf("%d",f(6)) | cout<<f(6)
- d. printf("%d",f(7)) | cout<<f(7)

3. Se consideră următorul subprogram recursiv:

```
void p(int i,int n)
{int j;
  if(i<=n)
    {p(i+1,n);
     for(j=1;j<=i;j++)
       cout<<j; | printf("%d",j);
     cout<<endl; | printf("\n");
    }
}
```

Ce se va afișa în urma apelului p(1,3)?

- | | | | |
|--------|--------|--------|------|
| a. 123 | b. 123 | c. 321 | d. 1 |
| 23 | 12 | 32 | 12 |
| 3 | 1 | 3 | 123 |

6. Se consideră subprogramul recursiv definit alăturat. Ce se va afișa în urma apelului f(1,3)?

```
void f(int i,int n)
{if(i<=n)
  {cout<<"*"; | printf("*");
   f(i+1,n);
   cout<<"#"; | printf("#");}
}
```

- | | | | |
|----------|----------|----------|----------|
| a. ##### | b. ##### | c. ##### | d. ##### |
|----------|----------|----------|----------|

8. Se consideră următorul subprogram recursiv:

```
long f(int x,int n)
{if(n==0) return 1;
  else if (n%2==0) return f(x,n/2)*f(x,n/2);
  else return x*f(x,n-1);}
```

Ce valoare va întoarce subprogramul după apelul $f(2,10)$?

- a. 10^2 b. 2^{10} c. 20 d. 2^9

5. Pentru definiția alăturată a subprogramului f , ce se va afișa la apelul $f(27524)$?

```
void f(int n){
  if(n){
    if(n%2){
      printf("%d",n%10);/cout<<n%10;
      f(n/10);}
    else{f(n/10);
      printf("%d",n%10);/cout<<n%10;}
  }
}
```

- a. 27524 b. 75422 c. 57224 d. 42572

4. Știind că tabloul b este declarat prin $\text{int } b[5]=\{1,2,3,4,5\}$; stabiliți ce se afișează în urma apelului $f(b,m,m-1)$, a funcției f cu definiția alăturată, în care m este egal cu 5?

```
void f(int a[5],int n,int i)
{if (i>=0)
{cout<<a[i]%10;/printf("%d",a[i]%10);
  f(a,n,i-1);
  if (a[i]<9)
  cout<<i; /printf("%d",i); }}
c. 1234512345                      d. 5432101234
```

- a. 1234554321 b. 5432154321

6. Se consideră definiția alăturată a subprogramului f . Ce se va afișa în urma apelului $f(13)$?

```
void f(int n)
{ if (n>0) f(n/10);
  printf("%d",n%10);/cout<<n%10;
}
```

- a. 013 b. 310 c. 31 d. 13

6. Se consideră subprogramul recursiv cu definiția alăturată. În urma apelului $p(n,2)$ funcția returnează valoarea 1 dacă și numai dacă:

```
int p(unsigned int n,
unsigned int x)
{if (n<x) return 0;
  if (x==n) return 1;
  else
  if (n%x==0) return 0;
  else return p(n,x+1);
}
```

- a. n este un număr natural impar b. n este un număr natural neprim
c. n este un număr natural par d. n este un număr natural prim

3. Se consideră subprogramul recursiv cu definiția alăturată :

```
void p(int n)
{ if (n!=1)
  { printf("%d ",n); / cout<<n<<" ";
    if(n%2==0) p(n/2);
    else p(3*n+1);
  }
else printf("%d",1); / cout<<1;
}
```

Ce valori vor fi afișate pe ecran în urma apelului $p(10)$?

- a. 5 16 8 4 2 1 b. 10 5 16 8 4 2 1
c. 10 5 16 8 4 2 d. 10 5 4 2 1

5. Se consideră subprogramul recursiv cu definiția alăturată:

```
void g(int n)
{int i;
  if(n>=1)
  {for(i=1;i<=n;i++)
    printf("%d ",n); / cout<<n<<" ";
    g(n-1);
  }
}
```

Ce valori vor fi afișate pe ecran în urma apelului $g(4)$?

- a. 4 4 4 4 3 3 3 2 2 1
c. 1 1 1 1 2 2 2 3 3 4

- b. 4 1 2 3 4 4 3 2 1 4
d. 4 3 2 1 3 2 1 2 1 1

6. Care trebuie să fie valoarea variabilei întregi v în apelul $F(v, s)$ dacă valoarea inițială a variabilei întregi s este 0, iar valoarea acesteia în urma executării subprogramului F , definit alăturat, este 4301?

```
void F(long int n, long int &s)
{  if (n > 0)
  {  s = s * 10 + n % 10;
    F(n / 10, s);
  }
}
```

- a. 1043 b. 1304

- c. 4301 d. 1034

5. Ce valoare va avea variabila întreagă s , în urma apelului $Suma(1, s)$, știind că variabila globală n are valoarea 10, variabila s are înainte de apel valoarea 0, iar subprogramul $Suma$ este definit alăturat?

```
void Suma(int i, int &s)
{ if (i <= n)
  {  if(i % 2)    s = s + 2;
    else        s = s - 1;
    Suma(i + 1, s);
  }
}
```

- a. 8 b. 5

- c. 7 d. 6

1. Câte caractere „*” se vor afișa în urma executării subprogramului alăturat, la apelul $Scrie(1,1)$?

```
void Scrie(int i, int j)
{if(i <= 3)
  if(j <= i)
  {  printf("*"); | cout<<'*';
    Scrie(i, j+1);
  }
  else Scrie(i+1, 1);
}
```

- a. 5 b. 6

- c. 3 d. 10

6. Ce se va afișa în urma apelului $F(123)$, pentru funcția F definită alăturat:

```
int F(int n)
{  if ( n == 0 || F(n / 10) )
  {  cout << n % 10;
    return 1;
  }
  return 0;
}
```

- a. 0123 b. 123

- c. 321 d. 3210

3. Se consideră funcția alăturată. Ce se va afișa pe ecran la apelul $f(13)$?

```
void f(int x)
{
  if (x>1) f(x/2);
  printf("%d",x%2); | cout<<x%2;
}
```

- a. 0101 b. 1100

- c. 1101 d. 1011

6. Ce se va afișa pe ecran la apelul `p(630,2,1)`?

```
void p(int x,int y,int k)
{if(x>1)
  if(x%y==0)
    {if(k==1)
      printf("%d ",y); | cout<<y<<" ";
      p(x/y,y,k+1);
    }
  else p(x,y+1,1);
}
```

- a. 2 b. 2 3 3 3 5 7 c. 2 3 5 7 d. 2 1 3 2 5 1 7 1

5. Se consideră următoarea funcție recursivă apelată numai pentru numere naturale nenule:

```
int f(int a, int b)
{ if (a<b) return a;
  else return f(a-b,b); }
```

Care din următoarele funcții este echivalentă cu funcția dată?

- a. `int f(int a, int b) { return a*b; }` b. `int f(int a, int b) { return a%b; }`
c. `int f(int a, int b) { return a-b+1; }` d. `int f(int a, int b) { return a/b; }`

3. Fie funcția recursivă:

```
int f(int i, int j)
{ if (i==1 || j==1) return 1;
  else if (i<j) return f(i,j-1)+f(i-1,j-1);
  else return f(i-1,j)+f(i-1,j-1);
}
```

Ce returnează funcția la apelul `f(2,3)`?

- a. 5 b. 3 c. 4 d. 2

3. Fie funcția `f` definită alăturat. Ce se afișează în urma apelului `f(523)`?

```
void f(int x)
{cout<<x%10; / printf("%d",x%10);
  if(x>9)
    f(x/10);
  cout<<x%10; / printf("%d",x%10);}
c. 325                                      d. 523523
```

- a. 325523 b. 325325

6. Se consideră subprogramul `f` cu definiția alăturată. Ce se afișează în urma apelului `f(6)`?

```
void f(int x)
{if (x>=0)
  {cout<<x; / printf("%d", x);
  f(x-1);
  if(x%2==0)
    cout<<x; / printf("%d", x);
  }}}
```

- a. 65432100246 b. 65432106420 c. 0123456 d. 6543210

2. Se consideră subprogramul recursiv alăturat. Ce se afișează la apelul `tip(4)`?

```
void tip(int i)
{
    printf("X");
    if (i>0)
        if (i%2==0) tip(i/2);
        else tip(i-1);
}
```

- a. XXXXXX b. XXXX c. XX d. XXX

2. Se consideră subprogramul `f` cu definiția alăturată și o variabilă `y` de tip `int`. Ce se va afișa în urma executării secvenței de program următoare:

```
y=0; f(4, y);
cout<<y; / printf("%d", y);
```

```
void f(int x, int&k)
{if (x>1)
    {k=k+1;
    f(x-1,k);}
}
```

- a. 4 b. 0 c. 5 d. 3

3. În subprogramul recursiv alăturat se consideră că `v` este un vector declarat global, format din numere întregi. Pentru care dintre vectorii `v` următori, expresia `ddd(0, 18)` returnează valoarea 9?

```
float ddd(int i, int j)
{
    if (v[i]==v[j]) return i;
    else return ddd(i+1, j-1);
}
```

- a. (1, 2, 1, 2, 3, 1, 2, 3, 4, 1, 1, 2, 3, 4, 1, 2, 3, 1, 2)
b. (1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0)
c. (1, 2, 1, 2, 3, 1, 2, 3, 4, 0, 4, 3, 2, 1, 3, 2, 1, 2, 1)
d. (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 2, 2, 2, 2, 2, 2, 2, 2)

4. În subprogramul recursiv alăturat se consideră că vectorul de numere întregi `v` este declarat global. Pentru care dintre vectorii `v` următori, la apelul `star(0)`, se afișează 7 asteriscuri?

```
void star(int i)
{
    if (i<10){
        printf("*");
        if (v[i]==i+1) star(i+2);
        else star(i+1);
    }
}
```

- a. (1, 4, 3, 2, 1, 6, 5, 4, 3, 10) b. (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
c. (3, 2, 1, 4, 3, 6, 7, 2, 9, 2) d. (10, 9, 8, 7, 6, 5, 4, 3, 2, 1)

6. Pentru definiția alăturată a subprogramului `f`, stabiliți ce valoare returnează apelul `f(6, 3)`?

```
long f(int n, int k)
{long s=0, i;
    if (n==k || k==1) return 1;
    if (n<k) return 0;
    for(i=1; i<=k; i++) s = s + f(n-k, i);
    return s;
}
```

- a. 3 b. 1 c. 2 d. 4