

- Să se scrie funcția pentru aproximarea valorii unei integrale, definită prin funcția $f(x)$, pe un interval dat, prin metoda trapezelor.

Funcția are ca parametri de intrare capetele intervalului pe care este definită integrala (a și b), numărul de diviziuni ale intervalului (n) și adresa funcției care se integrează (f). Funcția returnează, prin numele ei, valoarea aproximativă a integralei. Cu cât numărul de diviziuni este mai mare (lungimea unui subinterval mai mică) cu atât mai bună este aproximarea.

```
double trapez(double a,double b,int n,double (*f)(double))
{ double h,i;
  int j;
  h=(b-a)/n; i=0.0;
  for(j=0;j<=n;j++)
    i+=(*f)(a+j*h);
  i*=h; return i;}
```

- Să se scrie funcția pentru determinare celui mai mare divizor comun dintre 2 numere naturale.

Funcția are ca parametri de intrare cele două numere (a și b) și returnează, prin numele ei, valoarea celui mai mare divizor comun.

- *varianta recursivă:*

```
long cmmdc(long a,long b)
{ long c;
  if(a==b) c=a;
  else if(a>b) c=cmmdc(a-b,b);
    else c=cmmdc(a,b-a);
  return c;}
```

- *varianta iterativă:*

```
long cmmdc(long a,long b)
{ long r,d=a,i=b;
  do {r=d%i; d=i; i=r;}
  while(r!=0);
  return d;}
```

- Să se scrie o funcție eficientă pentru ridicarea unui număr la o putere naturală.

Funcția are ca parametri baza (b) și exponentul (e) și returnează, prin numele ei, valoarea cerută.

-*varianta iterativă:*

```
long putere(int b,int e)
{ long p=1;
  while(e)
    if(e%2) {p*=b;e--;}
    else {b*=b; e/=2;}
  return p;}
```

-*varianta recursivă:*

```
long putere(int b,int e)
{ long p;
  if(!e) p=1;
  else if(e%2) p=b*putere(b,e-1);
    else p=putere(b,e/2)*putere(b,e/=2);
  return p;}
```