

Memorator funcții predefinite C++

Funcții matematice

Funcția	Prototipul funcției	Fișierul antet	Semnificația rezultatului	Observații
pow(x,y)	double pow (double x,double y); long double pow (long double x, long double y); float pow (float x,float y); double pow (double x, int y); long double pow (long double x, int y);	<cmath>	- Rezultatul este x ridicat la puterea y: x^y - dacă $x < 0$, y trebuie să fie întreg	
sqrt(x)	double sqrt (double x); float sqrt (float x); long double sqrt(long double x);	<cmath>	- Rezultatul este radical din x, $x \geq 0$	Atenție In C există doar varianta double a funcției și are același nume
ceil(x)	double ceil (double x); float ceil (float x); long double ceil (long double x);	<cmath>	Rezultatul este cel mai mic întreg mai mare sau egal cu x Ex: ceil(2.4) este 3.0 ceil(2.8) este 3.0 ceil(-2.4) este -2.0 ceil(-2.8) este -2.0	
floor(x)	double floor (double x); float floor (float x); long double floor (long double x);		Rezultatul este cel mai mare întreg mai mic sau egal cu x Ex: floor (2.4) este 2.0 floor (2.8) este 2.0 floor (-2.4) este -3.0 floor (-2.8) este -3.0	
fabs(x)	double fabs (double x); float fabs (float x); long double fabs (long double x);	<cmath>	Rezultatul este valoarea absolută a lui x	

Memorator funcții predefinite C++

abs(x)	double abs (double x); float abs (float x); long double abs (long double x);	<cmath>	• Rezultatul este valoarea absolută a lui x	Atenție In C, abs este declarată doar în <cstdlib> (și operează doar pe valori întregi).
labs(n)	long int labs (long int n);	<cstdlib>	• Rezultatul este valoarea absolută a lui n	
sin(x)	double sin (double x); float sin (float x); long double sin (long double x);	<cmath>	• Rezultatul este sinusul valorii x • x este un unghi exprimat în radiani	
cos(x)	double cos (double x); float cos (float x); long double cos (long double x);	<cmath>	• Rezultatul este cosinusul valorii x • x este un unghi exprimat în radiani	
tan(x)	double tan (double x); float tan (float x); long double tan (long double x);	<cmath>	• Rezultatul este tangenta valorii x • x este un unghi exprimat în radiani	
asin(x)	double asin (double x); float asin (float x); long double asin (long double x);	<cmath>	• Rezultatul este unghiul exprimat în radiani al cărui sinus este x • x este o valoare reală din [-1,+1].	
acos(x)	double acos (double x); float acos (float x); long double acos (long double x);	<cmath>	• Rezultatul este unghiul exprimat în radiani al cărui cosinus este x • Rezultatul aparține intervalului [-pi/2,+pi/2] • x este o valoare reală din intervalul [-1,+1].	
atan(x)	double atan (double x); float atan (float x); long double atan (long double x);	<cmath>	• Rezultatul este unghiul exprimat în radiani a cărui tangentă este x • Rezultatul aparține intervalului [-pi/2,+pi/2]	

Memorator funcții predefinite C++

log(x)	double log (double x); float log (float x); long double log (long double x);	<cmath>	• Rezultatul este logaritm natural din x, • x>0	
log10(x)	double log10 (double x); float log10 (float x); long double log10(long double x);	<cmath>	• Rezultatul este logaritm în baza 10 din x, • x>0	
exp(x)	double exp (double x); float exp (float x); long double exp (long double x);	<cmath>	• Rezultatul este valoarea funcției exponențiale pt. x	
fmod(x,y)	double fmod (double x, double y); float fmod (float x, float y); long double fmod (long double x, long double y);	<cmath>	• Rezultatul este restul împărțirii lui x la y rezultat = x - cat * y Ex: fmod(5.3,2)= 1.300000 fmod(18.5,4.2)= 1.700000	Operatorul % este doar pentru numere întregi

Memorator funcții predefinite C++

Funcții pentru numere aleatoare

Funcția	Prototipul funcției	Fișierul antet	Semnificația rezultatului	Observații
srand (s);	void srand (unsigned int seed);	<cstdlib>	Inițializează generatorul de numere aleatoare	Se folosește de obicei: srand (time(NULL));
rand()	int rand (void);	<cstdlib>	- Rezultatul este un număr pseudoaleator din intervalul [0,RAND_MAX]. - RAND_MAX este o constantă definită în <cstdlib> , valoarea sa este dependentă de implementare, dar are cel puțin valoarea 32767 - Ex de folosire a funcției rand: v1=rand()%100; // v1 are valori între 0 și 99 v2 = rand()%100 + 1; // v2 are valori între 1 to 100 v3= rand() % 30+1985; // v3 are valori între 1985-2014	Exemplu de generare a unui număr aleator: <pre>#include <iostream> #include <stdlib.h> #include <time.h> using namespace std; int main () { srand (time(NULL)); cout<<rand()%100; return 0; }</pre>

Memorator funcții predefinite C++

Funcții de conversie

Funcția	Prototipul funcției	Fișierul antet	Semnificația rezultatului	Observații
atof(str)	<code>double atof (const char * str);</code>	<code><cstdlib></code>	- Rezultatul este conversia unui șir de caractere într-o valoare double. - Sunt convertite caracterele din șir, începând cu primul, până la acel caracter care nu corespunde sintaxei unui număr real de acolo încolo caracterele sunt ignorate. Ex: <code>atof("12.34fg") = 12.34</code> - Dacă primul caracter din șir nu corespunde unei valori double, rezultatul este 0 Ex: <code>atof("w1.2") = 0</code>	
atoi(str)	<code>int atoi (const char * str);</code>	<code><cstdlib></code>	- Rezultatul este conversia unui șir de caractere într-o valoare întreagă. Ex: <code>atoi("123") = 123</code> - Sunt convertite caracterele din șir, începând cu primul, până la acel caracter care nu corespunde sintaxei unui număr întreg, de acolo încolo caracterele sunt ignorate. Ex: <code>atoi("12q6") = 12</code> - Dacă primul caracter din șir nu corespunde unei valori int, rezultatul este 0 Ex: <code>atoi("w12") = 0</code>	

Memorator funcții predefinite C++

atol(str)	long int atol (const char * str);	<cstdlib>	Analog cu atoi, doar că rezultatul este de tip <i>long int</i>	
itoa(v,str,b);	char * itoa(int v, char * str, int b);	<stdlib.h>	• Convertește valoarea întreagă v într-un șir de caractere str, conversia se face în baza b • Dacă baza b este 10 și valoarea v este negativă atunci șirul rezultat va începe cu semnul minus(-); pentru orice altă bază b valoarea v va fi considerată întotdeauna unsigned • str trebuie să fie suficient de mare pentru a putea păstra șirul rezultat • valoarea returnată este un pointer la un șir ce are același conținut ca și str	Ex: <pre>#include <stdlib.h> #include<iostream> using namespace std; int main () { int i=15; char s [33]; itoa (i,s,10); cout<<"zecimal: "<<s<<endl; itoa (i,s,16); cout<<"hexa: "<<s<<endl; itoa (i,s,2); cout<<"binar: "<<s<<endl; return 0; }</pre>
tolower(c)	int tolower (int c);	<cctype>	• Modifică o literă mare (A-Z) într-o literă mică (a-z) • Dacă c nu este o literă mare se returnează c nemodificat	
toupper(c);	int toupper (int c);	<cctype>	• Modifică o literă mică (a-z) într-o literă mare (A-Z) • Dacă c nu este o literă mică se returnează c nemodificat	

Memorator funcții predefinite C++

Funcții referitoare la fișiere

Funcția	Prototipul funcției	Fișierul antet	Semnificația rezultatului
remove(ume_fișier)	<pre>int remove (const char * filename);</pre>	<cstdio>	• Șterge un fișier specificat prin nume sau cale. • Fișierul trebuie închis înainte de ștergere. • Dacă ștergerea s-a efectuat atunci întoarce 0, altfel o valoare diferită de 0 Ex: <pre>#include <iostream> using namespace std; int main () { if(remove("date.in") != 0) cout<<"Eroare la ștergere"<<endl; else cout<< "Fișierul a fost șters" ; return 0;}</pre>
rename(ume_vechi, ume_nou)	<pre>int rename (const char * oldname, const char * newname);</pre>	<cstdio>	• Redenumeste un fișier. • Dacă redenumirea s-a efectuat atunci întoarce 0, altfel o valoare diferită de 0 Ex: <pre>#include <iostream> using namespace std; int main () { int result;char oldname[] ="date.in", newname[] ="date.txt"; result= rename(oldname , newname); if (result == 0) cout<<"Fișier redenumit cu succes"<<endl; else cout<< "Eroare la redenumire" ; return 0;}</pre>

Memorator funcții predefinite C++

Alte funcții

Funcția	Prototipul funcției	Fișierul antet	Semnificația rezultatului
exit()	<code>void exit (int status);</code>	<cstdlib>	• Termină imediat execuția programului indiferent de funcția care apelează exit