

1. C realizuoja vienintelį parametrų perdavimo mechanizmą - pagal reikšmę (angl. *by value*; Paskalyje parametrai-reikšmės). Šis perdavimo būdas pasižymi tuo, kad į funkciją yra perduodamos argumentų kopijos Tiksliau būtų sakyti, kad funkcijos lokalūs kintamieji (t.y. formalūs parametrai) yra inicijuojami argumentų (faktinių parametrų) reikšmėmis. Todėl naudojant šį mechanizmą patys argumentai funkcijos aplinkoje nėra keičiami.
2. Kitas (plačiąja informatikos prasme kiti) perdavimo mechanizmas - pagal nuorodą (angl. *by reference*; Paskalyje parametrai-kintamieji) gali būti realizuojami (imituojami) parametrų perdavimo pagal reikšmę mechanizmo pagalba.
3. Techniškai argumentų perdavimo pagal nuorodą imitavimą, turint perdavimą pagal reikšmę, galima išskaidyti į 3 žingsnius:
  - 1) parametrų-reikšmių tipus padarome rodykliniais (tiksliau - pridėdam vieną rodyklės deklaratorių), taip paruošdami juos adresinėms reikšmėms (tiksliau – pasakydami, kad parametrai manipuliuos su to tipo kintamųjų adresais);
  - 2) visur kur naudojami parametrai-reikšmės pritaikome rodyklės (išadresavimo) operaciją;
  - 3) kreipinyje į funkciją kaip argumentus perduodame kintamųjų adresus.

Perdavimas pagal reikšmę – čia netinkamas	Imituojamas perdavimas pagal nuorodą	C++ perdavimas pagal nuorodą
<pre>void swap_blogas(int x,int y) {     int temp;     temp=x;     x=y;     y=temp; }  int main() {     int a=1, b=2;     swap_blogas(a,b);     /* a=1,b=2 */     ... }</pre>	<pre>void swap(int* x,int* y) {     int temp;     temp=*x;     *x=*y;     *y=temp; }  int main() {     int a=1, b=2;     swap(&amp;a,&amp;b);     /* a=2,b=1 */     ... }</pre>	<pre>void swap_cpp(int&amp; x,int&amp; y) {     int temp;     temp=x;     x=y;     y=temp; }  int main() {     int a=1, b=2;     swap_cpp(a,b);     /* a=2,b=1 */     ... }</pre>

4. C++ turi tikrąjį parametrų perdavimo pagal nuorodą mechanizmą. Jis realizuojamas pasitelkiant naują C++ savybę (konstrukciją) – nuorodas (angl. *references*), kurias apibendrintai galima vadinti kintamųjų (objektų) pseudonimais (angl. *aliases*).