

Obsah

1. Úvod	6
2. C++ - seznamte se	7
2.1. Užití proudů pro vstup a výstup	7
2.2. C++ komentáře	8
2.3. Funkční prototypy	8
2.4. Otevřená pole	9
3. C++ rozšíření	10
3.1. Implicitní parametry funkcí	10
3.2. Umístění deklarací proměnných	10
3.3. Operátor rozlišení platnosti proměnné	11
3.4. Inline funkce	12
3.5. Kvalifikátor const	13
3.6. Výčtové typy	14
3.7. Přetížené funkce	14
3.8. Volací konvence	15
3.8.1. Volací konvence cdecl	15
3.8.2. Volací konvence pascal	16
3.8.3. Konvence rychlého volání _fastcall	16
3.8.4. Konvence volání Object Data	16
3.9. Odkazy	16
3.9.1. Inicializace odkazu	16
3.9.2. Odkazy a ukazatelé	17
3.9.3. Odkazy jako funkční parametry	17
3.9.4. Odkaz jako návratová hodnota	18
4. Třídy	20
4.1. Založení nového datového typu	20
4.1.1. V jazyce C	20
4.1.2. V jazyce C++	21
4.1.3. Použití třídy	21
4.2. Členové třídy	22
4.2.1. Viditelnost členů třídy	22
4.2.2. Členské metody	22
4.2.3. Konstruktor	23
4.2.4. Destruktor	23
4.3. Zakládání a rušení objektu	24
4.4. Přístup k datovým členům	24
4.4.1. Přístupové funkce a veřejné datové členy	26
4.4.2. Návratová hodnota odkazem	26
4.5. Konstantní objekty a členské metody	27
4.6. Členské objekty	28
5. Třídy a dynamická alokace paměti	30

5.1. Volný paměťový prostor	30
5.1.1. Operátor new	30
5.1.2. Operátor delete	31
5.1.3. Volný prostor a předdefinované typy	31
5.2. Třídy obsahující ukazatele	32
5.3. Přiřazovací operátor	34
5.4. Ukazatel <code>this</code>	35
5.4.1. Použití výrazu <code>*this</code> jako návratové hodnoty funkce	36
5.5. Přiřazení a inicializace	36
5.5.1. Copy konstruktor	37
5.5.2. Předávání a vracení objektů	38
5.5.3. Předávání a vracení referencí na objekty	39
5.6. Práce s operátory <code>*</code> a <code>-></code>	40
6. Další vlastnosti tříd	42
6.1. Statické členy tříd	42
6.1.1. Statické datové členy	42
6.1.2. Statické členské funkce	43
6.2. Přátelské třídy	44
6.2.1. Přátelské funkce	47
6.3. Pole objektů	48
6.3.1. Volný prostor a pole objektů	48
6.4. Pokročilé techniky v paměťovém prostoru	50
6.4.1. Funkce <code>set_new_handler</code> (<code>_set_new_handler</code> MICROSOFT C)	50
6.4.2. Přetížení operátorů <code>new</code> a <code>delete</code>	51
6.4.3. Operátory <code>new</code> a <code>delete</code> specifické pro třídu	52
7. Dědičnost a polymorfismus	56
7.1. Příbuzné typy v jazyce C	56
7.2. Práce s příbuznými typy v jazyce C++	58
7.2.1. Předefinování členů základní třídy	61
7.2.2. Odvozené konstruktory tříd	62
7.2.3. Konverze mezi základními a odvozenými třídami	63
7.2.4. Kolekce používající ukazatele na základní třídy	64
7.3. Virtuální funkce	65
7.3.1. Polymorfismus	67
7.3.2. Dynamické vazby	67
7.3.3. Implementace virtuálních funkcí	68
7.3.4. Čisté virtuální funkce	69
7.3.5. Destruktoři v základních a odvozených třídách	71
7.4. Chráněné datové členy (Protected members)	71
7.5. Veřejné a soukromé základní třídy	72
7.6. Vícenásobná dědičnost	73
7.7. Struktury a uniony v jazyce C++	74
8. Přetížení operátorů a konverzních funkcí	75
8.1. Přetěžování operátorů	75
8.1.1. Pravidla pro přetěžování operátorů	76
8.1.2. Kdy nepřetěžovat operátory	77
8.2. Operátory přetěžitelné pouze jako členské metody	78

8.3. Přetěžování operátorů numerických tříd	78
8.3.1. Definování operátorů jako přátelských funkcí	80
8.3.2. Prefixové a postfixové operátory ++ a --	81
8.3.3. Přetěžování přiřazovacích operátorů typu +=	82
8.3.4. Typy pro přetěžování aritmetických operátorů	82
8.4. Přetěžování operátorů pro třídu pole	82
8.5. Konverze tříd	85
8.5.1. Konverze konstruktorem	85
8.5.2. Konverzní operátory	86
8.5.3. Nejasnosti při použití konverzí a konverzních operátorů	87
8.5.4. Nejasnosti mezi vzájemnými konverzemi dvou tříd	88
9. Rejstřík	90
10. Literatura	92