

Opakování „programování“ – předchozí kurzy

HW návaznost

- procesor – sběrnice, instrukční sada, optimalizace rychlosti, datové typy, operace (matematické, logické, podmínky, skoky, podprogram ...)
- paměti a periferie
- adresování

Tvorba programu

- návrh
- kritéria hodnocení

Programové prostředky (editor, překladače, ladicí prostředky, sestavení programu)

Jazyk

- klíčová slova
- datové typy
- základní mechanizmy jazyka

Processor

- má určitý počet instrukcí (příkazy ve strojovém kódu)
- instrukce říká, co a s čím se má udělat
- instrukce trvá určitý počet cyklů (času, přístupu k paměti ...)
- obsahuje registry (vnitřní paměti)
- akumulátor – výpočetní jednotka (ALU)
- je schopen pracovat s určitými datovými typy
- čítač instrukcí říká, kde leží další instrukce (ovlivňují ho instrukce skoků (podmíněné/nepodmíněné)), cykly
- podprogram (call/return) – zásobník
- interrupt (přerušování) - volatile proměnné
- registr příznaků – výsledky operací (nulovost, kladnost, přetečení ...)
- synchronizační mechanismy a instrukce pro spolupráci více procesorů

Paměť

- v paměti jsou uloženy instrukce a data programu
- program obsahuje instrukce, které se vykonávají
- datová oblast příslušná programu – základní data pro proměnné programu
- zásobník – lokální data, adresy při podprogramech
- statické a globální proměnné v datové části programu (inicializace)
- „volná“ datová oblast – je možné o paměť z ní požádat „systém“
- mapování periférií do paměti – data se mění "nezávisle" – volatile proměnné
- cache paměť na čipu – podstatně rychlejší přístup k datům (dnes několika úrovně – jádro, procesor, chipset)

Datové typy (vázané na procesor, nebo emulované v SW)

- celočíselné – znaménkové x bezznaménkové (zápis binárně, oktalově, dekadicky, hexadecimálně)
- s desetinnou čárkou
- podle typu procesoru a registru (spojení registrů) je dána přesnost (velikost typu v bytech)
- adresa x ukazatel
- pro adresování (segment:offset, indexovaný přístup ...)
- pro vyjádření znaku se využívá celočíselná proměnná – teprve její interpretací (například na tiskárně) „vznikne“ znak.
- Základní znaková sada (ASCII, EBCDIC) je osmibitová
- Rozšířená znaková sada UNICODE
- znakové sady s konstantním nebo proměnným počtem bytů na znak

Matematické operace

- Sčítání, odčítání – základ (celočíselné)
- Násobení, dělení
- Mocniny, sinus, cos, exp ... jsou většinou řešeny podprogramy, nebo pomocí tabulek (a interpolací). Jsou součástí knihoven ne jazyka.

Boolovské operace

- použití pro vyhodnocování logických výrazů
- Tabulka základních logických funkcí pro kombinace dvou proměnných

První dva řádky tabulky ukazují možné varianty/kombinace proměnných A a B.

Další řádky ukazují všechny možné výsledky vstupních kombinací.

Každý řádek je jedna operace nad vstupními proměnnými.

Ve sloupci jsou vstupní hodnoty a příslušná hodnota výsledku pro danou operaci.

0	0	1	1	vstup A
0	1	0	1	vstup B
0	0	0	0	nulování
0	0	0	1	AND
0	0	1	0	přímá inhibice (negace implikace) - Nastane-li A, nesmí nastat B.
0	0	1	1	A
0	1	0	0	zpětná inhibice
0	1	0	1	B
0	1	1	0	XOR nonekvivalence (jsou-li proměnné různé je výsledkem 1, jsou-li stejné, pak 0)
0	1	1	1	OR
1	0	0	0	negace OR
1	0	0	1	negace XOR (výsledek je 1, pokud jsou proměnné stejné, pokud jsou různé pak je výsledek 0)
1	0	1	0	negace B
1	0	1	1	zpětná implikace
1	1	0	0	negace A
1	1	0	1	přímá implikace (nastane-li stav A, je výsledek řízen stavem B. Z nepravdy A nemůžeme usoudit na stav B – mohou být platné oba stavy (nebude-li přšet, nezmoknem). Pokud platí A je možné z výsledku usuzovat na B (B je stejné jako výsledek) pokud A neplatí nelze o vztahu výsledku a B nic říci.
1	1	1	0	negace AND
1	1	1	1	nastavení do jedničky

Způsoby „adresování“

- Součást instrukce - INC A (přičti jedničku k registru A) – registr, se kterým se pracuje je přímo součástí instrukce
- Přímý operand – JMP 1234 – skoč na danou adresu – je uvedena v paměti za instrukcí. Může mít i relativní formu k současné pozici
- Adresa je uvedena jinde (v jiné proměnné) – PUSH B – registr B se uloží na zásobník, LD A, (BC) – do registru A se načte hodnota z adresy ve dvojici registrů BC
- Indexové adresování MOVIX A,BC,IX – do registru A se načte hodnota z paměti, která je posunuta o IX (index) od adresy v registru BC (báze). Registry BC, IX bývají pevně určené (tj. není možné určit, který registr obsahuje bázi a který index/offset)

Programování

- Rozbor úlohy – které funkce patří k sobě (knihovny), rozhraní funkcí (předávané a návratové hodnoty), datové typy pro proměnné
- Algoritmy – řešení daného úkolu ve funkci
- Zapsání kódu
- překlad – „jazyková“ správnost
- Ladění kódu – debugging – „funkční“ správnost
- Testovací databáze

Postup programování

- požadované vlastnosti
- návrh činnosti
- návrh datových struktur
- návrh funkčních volání

Hodnocení programu

- Výkon a efektivita – čas, využití zdrojů
- Spolehlivost – HW, SW (na podněty musí správně reagovat)
- Robustnost – odolnost proti „rušení“, chybovým nebo neočekávaným stavům (HW, SW, uživatel)
- Použitelnost – jak je „příjemný“ pro uživatele, jak snadno se zapracovává do programu
- Přenositelnost – jak velké úpravy je nutné dělat při překladu na jiné platformě (jiným překladačem) – jazyk, použité funkce, návaznost na OS, velikost datových typů, endiány ...
- Udržovatelnost – dokumentace, komentáře, přehlednost
- Kultura programování – programátorský styl, komentáře (popisují proč je to tak), dokumentace

Programovací prostředí

- Editor – vytvoření zdrojových a hlavičkových souborů (co to je, jaká je mezi nimi vazba)
- Překladač + preprocesor – direktivy preprocesoru #xxx, překlad do mezikódu, kontrola syntaktických chyb
- Linker – spojení částí programu (.o, .obj, .lib, .dll, ...) do jednoho celku (.exe, .lib, .dll, ...)
- knihovny (.c, .cpp, .o, .obj, .lib, .dll) předpřipravené části kódu, které zjednodušují psaní programu. Jejich rozhraní je oznámeno v hlavičkovém souboru.
- Debugger – je možné hledat chyby v programu. Trasování – procházení programu po krocích nebo částech s možností zobrazení hodnot proměnných nebo paměťových míst

- Projekt – sada souborů, jejichž zpracováním vznikne výsledek (.exe, .dll, ...)
- Řešení (solution) - sada společných projektů
- Překlad – kompilace (zpracování zdrojových souborů); linkování (sestavení programu), build (kompilace změněných souborů a linkování); build all, rebuild (kompilace všech souborů a linkování); reset, clean, clear (smazání všech souborů (meziproduktů) překladu)

Opakování jazyka C

Imperativní programování – popisujeme kroky, které má program vykonat

Strukturovanost programu – „grafická“ v rámci funkcí, programátorský styl, (firemní) kultura programování, program realizován pomocí funkcí (předávání parametrů),

Klíčová slova - cca 37 klíčových slov

void
char, short (int), int, long (int)
signed, unsigned
float, double, (long double)
union, struct, enum
auto, register, volatile, const, static
extern, typedef
sizeof
if, else, switch, case, default, break – (podmíněné větvení)
goto
return
for, while, do, continue, (break) - cykly a skoky
operátory (matematické a logické , přiřazení (možnost zřetězení), ternární operátor)
restrict, inline, _Bool, _Complex, _Imaginary (C99)

Datové typy

- datové typy – udávají přesnost, typ, znaménko, modifikátory
- velikost vázána na platformu (sizeof)
- celočíselné neznaménkové – pro logické operace (bitové, posuny...)
- složené datové typy (struktury, union)
- ukazatel – adresa spojená s typem, který na ní leží, ukazatelová aritmetika
- pole – návaznost na ukazatel, řetězce C (typ string)
- alokace paměti – automatické proměnné, dynamické proměnné (kde leží), globální proměnné, definice a deklarace (inicializace)
- výčtový typ enum

- psaní konstant (znaková 'a'; celočíselná -12, 034, 0xFA8ul; neceločíselné 23.5, 56e-3; pole "ahoj pole"
- escape sekvence \n \r \t \a \0 \0x0D
- konverze datových typů implicitní a explicitní
- typedef

Literál – jednoduchá, pevně daná přímo vyjádřená hodnota (12, 'a', "abc", {12,13,147}), kterou lze inicializovat proměnná (včetně pole, struktury...)

Boolovská logika

- použití logické (proměnná je brána jako celek nula/nenula) x matematické (po bitech)
- využití pro maskování
- spojeno s neznaménkovými celočíselnými typy
- hodnoty používané k vyjádření logické proměnné
- operace bit po bitu (nad bitovými řezy, bitwise) a s celým číslem (konverzí na bool)

Funkce

- návratová hodnota – jak se předává
- parametry funkce – lokální parametry, předávání hodnotou (využití ukazatele), předávání polí (v závislosti na definici)
- lokální parametry funkcí
- funkce main – musí mít návratovou hodnotu int, může mít parametry
- funkce pro (formátovaný) vstup a výstup – standardní vstupy a výstupy stdin, stdout, stderr;