

Zadání testu:**Jmeno:**

Napište funkci, která bude rozdělovat řetězce podle daného znaku. Vícenásobným voláním této funkce vytvořte několika řetězců podle daného oddělovače.

Obecná pravidla:

- testy provádějte v uvedeném pořadí
- manipulaci s pamětí provádějte až po provedení všech testů pro vstupní parametry
- chyby jsou signalizovány písmenem (V=hodnota, M=paměť) a číslem cyklu, ve kterém k chybě došlo.
- chyby hlásí checker (paměť a soubor – při nevhodné manipulaci ihned, jinak dává sumář na konci programu)
- pokud program spadne, jedná se většinou o špatnou manipulaci s pamětí (je nutné určit cyklus, při kterém k chybě došlo a vyzkoušet jaké parametry funkce dostala a jak na ně reagovala – trasování)

Návod:

- 1) Své funkce pište v souboru funkce2a.c s příslušným hlavičkovým souborem funkce2a.h. Hlavičkový soubor ošetřete. Volání svých funkcí provádějte z funkce Testování (tu překopírujte ze souboru s main). Po odladění svou funkci zaregistrujte způsobem, který je ukázán v souboru s main (registrujete funkci a druhý parametr je konstanta nadefinovaná v souboru test.h, která má ve svém názvu název registrované funkce).
- 2) Napište funkci Oddel, která oddělí část řetězce, který je jí předán jako parametr (aRet). Oddělovačem bude znak středník. Oddělovat se bude od konce textu (tj. najde se poslední středník v řetězci (první od konce) a vše včetně tohoto středníku se oddělí). Nebude-li oddělovač přítomen, oddělí se celý obsah vstupního řetězce. Pro oddělenou část se naalokuje paměť do které se zkopíruje.
Oddělený řetězec se ze vstupního odebere tak, že se pouze přesune znak konce řádku (nepřealokovává se). Výsledek předejte pomocí místa, na které ukazuje druhý z ukazatelů (aVystup). Výsledkem bude nový (naalokovaný) C řetězec jehož obsahem bude oddělený řetězec včetně oddělovacího znaku.
Prototyp funkce je int Oddel (char *aRet, char **aVystup)

Pro vstup: "Ahoj.; Jak.; Se.; Mas." - oddělený řetězec bude "; Mas." a původní řetězec bude "Ahoj.; Jak.; Se."

Návratové hodnoty funkce Oddel:

nezáporné číslo - délka načteného řetězce (bez ukončovacího znaku)

(testy na chyby provádějte v níže uvedeném pořadí)

- 1 (kterýkoli) předaný ukazatel nemá platnou adresu (tj. obsahuje hodnotu NULL)
- 2 na adrese odkazované ukazatelem pro uložení výsledku jsou již data (je tam platná adresa)
- 3 chyba při alokaci paměti (nedostatek paměti)

- 3) Napište funkci Rozdel (bez parametrů), která zavolá funkci Oddel tolikrát, dokud nebude řetězec vrácený funkcí Oddel (jako druhý parametr) prázdný (tj. bude obsahovat řetězec nulové délky). Ukazatel na řetězec pro prvotní zpracování (první volání) vrátí funkce char * AllocGetData() (naalokuje paměť, do které vloží data pro zpracování - řetězec vrátí v návratové hodnotě a tímto předá funkci, která se stane jeho vlastníkem (odalokuje ho)). Zajistěte, aby byla tato funkce napsána korektně.
Pokud nastane chyba v této funkci, nebo ve funkci volané, odalokujte paměť a funkci ukončete.

Pozn.: Funkci Oddel můžete při psaní funkce podle bodu 3 (nikoli 2) nahradit předdefinovanou funkcí fpzOddel