Maticová kalkulačka

Programátorská dokumentácia

Dominik Bobák

5ZYP21

Súbory:

Main.c

Funkcie:

* bool (\*MainFun)(MATICA\*\* m1, MATICA\*\* m2) – smerník na funkciu, ktorá riadi program. Aplikácia beží pokiaľ funkcia vracia true.
  + MATICA\*\* m1 – adresa smerníka na maticu 1 s ktorou bude funkcia pracovať
  + MATICA\*\* m2 – adresa smerníka na maticu 2 s ktorou bude funkcia pracovať
* int main(int argc, char\* argv[]) – úvodné nastavenie programu
  + int argc – počet vstupných parametrov
  + char\* argv[] – zoznam vstupných parametrov

Matica.h, Matica.c

Definované typy:

* MAX\_ROZMER 10 – maximálny rozmer matice

Štruktúry:

* MATICA:
  + unsigned int riadky – počet riadkov matice
  + unsigned int stlpce – počet stĺpcov matice
  + double\*\* hodnoty – dvojrozmerné pole reálnych hodnôt

Verejné funkcie:

* MATICA\* VytvorMaticu(unsigned int m, unsigned int n) – Vytvorí maticu MxN prvkov a vráti jej adresu.
  + unsigned int m – počet riadkov
  + unsigned int n – počet stĺpcov
* MATICA\* Scitanie(MATICA\* m1, MATICA\* m2) – Ak sú matice rovnakého tvaru tak matice sčíta, hodnotu uloží do novej matice a vráti jej adresu. Inak vráti NULL.
  + MATICA\* m1 – adresa matice, sčítanec
  + MATICA\* m2 – adresa matice, sčítanec
* MATICA\* Odcitanie(MATICA\* m1, MATICA\* m2) – Ak sú matice rovnakého tvaru tak matice odčíta, hodnotu uloží do novej matice a vráti jej adresu. Inak vráti NULL.
  + MATICA\* m1 – adresa matice, menšenec
  + MATICA\* m2 – adresa matice, menšiteľ
* MATICA\* Sucin(MATICA\* m1, MATICA\* m2) – Ak sa počet stĺpcov matice 1 rovná počtu riadkov matice 2 tak ich vynásobí, hodnotu uloží do novej matice a vráti jej adresu. Inak vráti NULL.
  + MATICA\* m1 – adresa matice, činiteľ
  + MATICA\* m2 – adresa matice, činiteľ
* MATICA\* SkalarnySucin(MATICA\* m, double cislo) – Každý prvok matice vynásobí zadaným čislom, výsledok uloží do novej matice a vráti jej adresu.
  + MATICA\* m – adresa matice ktorá sa bude násobiť
  + double cislo – číslo ktorým sa bude násobiť
* MATICA\* Transponovanie(MATICA\* m) – Transponuje maticu m, výsledok vloží do novej matice a vráti jej adresu.
  + MATICA\* m – adresa matice na transponovanie
* MATICA\* Inverzia(MATICA\* m) – Ak je matica štvorcového tvaru a jej determinant nie je rovný 0, vytvorí inverznú maticu a vráti jej adresu.
  + MATICA\* m – adresa matice na inverziu
* MATICA\* Jednotkova(unsigned int m) – Vytvorí maticu mxm, prvky na hlavnej diagonále naplní 1 a vráti jej adresu.
  + unsigned int m – počet riadkov a stĺpcov
* MATICA\* Skopiruj(MATICA\* m) – Hodnoty z matice m skopíruje do novej matice a vráti jej adresu.
  + MATICA\* m – adresa matice na skopírovanie
* void ZrusMaticu(MATICA\*\* m) – Vymaže maticu m z pamäte. Jej adresu nastaví na NULL
  + MATICA\*\* m – adresa smerníka na maticu
* void VypisMaticu(MATICA\* m) – Vypíše prvky matice na konzolu.
  + MATICA\* m – adresa matice na výpis
* double Determinant(MATICA\* m) – Vráti determinant matice m. Pokiaľ sa determinant nedá vypočítať vráti 0.
  + MATICA\* m – adresa matice pre výpočet determinantu
* bool ExistujeDeterminant(MATICA\* m) – Skontroluje či je možné vypočítať determinant matice m. Vráti true ak áno, false ak nie.
  + MATICA\* m – adresa matice pre kontrolu determinantu

Pomocné funkcie:

* double NasobRiadokStlpec(MATICA\* m1, MATICA\* m2, unsigned int m, unsigned int n) – Pomocná metóda pre násobenie matíc. Vráti hodnotu násobenia zvoleného riadku a stĺpca matíc.
  + MATICA\* m1 – adresa matice z ktorej sa zoberie zvolený riadok
  + MATICA\* m2 – adresa matice z ktorej sa zoberie zvolený stĺpec
  + unsigned int m – index riadku
  + unsigned int n – index stĺpca
* double PrvokInverzie(MATICA\* mPtr, int m, int n) – Pomocná metóda pre počítanie invertovanej matice. Vráti hodnotu determinantu matice mPtr zmenšenú o m-tý riadok a n-tý stĺpec. Ak je m+n nepárne číslo, hodnotu vynásobí -1.
  + MATICA\* mPtr – adresa matice pre výpočet
  + int m – index riadku ktorý sa nebude brať do úvahy
  + int n – index stĺpca ktorý sa nebude brať do úvahy
* void VymenPrvkyZoznamu(int\* zoznam, int index1, int index2, int n) – Pomocná metóda pre VypisZoznam a HodnotaDeterminantu. Vymení prvok v zozname na indexe 1 za prvok na indexe 2 a naopak.
  + int\* zoznam – adresa na zoznam prvkov
  + int index1 – index prvého prvku
  + int index2 – index druhého prvku
  + int n – počet prvkov v zozname
* void VypisZoznam(int\* zoznam, int i, int n, int pocetVymen) – Pomocná metóda pre počítanie determinantu. Vypíše prvky v zozname a všetky ich kombinácie s počtom výmen.
  + int\* zoznam – adresa zoznamu prvkov
  + int i – index posledného meneného prvku
  + int n – počet prvkov v zozname
  + int pocetVymen – počet doterajších výmen
* double HodnotaDeterminantu(int\* zoznam, int i, int n, int pocetVymen) – Pomocná metóda pre počítanie determinantu. Vynásobí medzi sebou všetky kombinácie prvkov a vráti ich súčet. Pokiaľ je počet doterajších výmen párny, hodnota sa vynásobí -1.
  + int\* zoznam – adresa zoznamu prvkov
  + int i – index posledného meneného prvku
  + int n – počet prvkov v zozname
  + int pocetVymen – počet doterajších výmen

Citac.h, Citac.c

Verejné funkcie:

* void NacitajMatice(const char\* nazovSuboru, MATICA\*\* m1, MATICA\*\* m) – Skúsi načítať matice zo súboru. Pokiaľ súbor neexistuje alebo je napísaný v zlom formáte, užívateľ musí matice nastaviť ručne.
  + const char\* nazovSuboru – názov súboru
  + MATICA\*\* m1 – adresa smerníka na prvú maticu
  + MATICA\*\* m2 – adresa smerníka na druhú maticu

Pomocné funkcie:

* bool TestujSubor(const char\* nazovSuboru) – Skontroluje či súbor existuje a je napísaný v správnom tvare. Ak áno vráti true, ink vráti false.
  + const\* char nazovSuboru – názov súboru
* void NacitajZoSuboru(const char\* nazovSuboru, MATICA\*\* m1, MATICA\*\* m2) – Načíta matice zo súboru.
  + const char\* nazovSuboru – názov súboru
  + MATICA\*\* m1 – adresa smerníka na prvú maticu
  + MATICA\*\* m2 – adresa smerníka na druhú maticu

Vstup.h, Vstup.c

Verejné funkcie:

* int NacitajCislo() – Načíta číslo z klávesnice.
* double NacitajDouble() – Načíta reálne číslo z klávesnice.
* MATICA\* NastavMaticu() – Nastaví maticu podľa parametrov z klávesnice.

Komunikator.h, Komunikator.c

Definované typy:

* SLEEPTIME 4000 – počet milisekúnd za ktorý počká program, kým sa vykoná ďalší príkaz.

Verejné funkcie:

* bool Komunikuj(MATICA\*\* m1, MATICA\*\* m) – Riadi beh programu. Vypýta si od užívateľa číslo operácie, ktorá sa má vykonať a vykoná ju. Vráti false ak sa užívateľ rozhodne program ukončiť. Inak vracia true.
  + MATICA\*\* m1 – adresa smerníka na maticu 1 s ktorou sa budú robiť operácie
  + MATICA\*\* m2 – adresa smerníka na maticu 2 s ktorou sa budú robiť operácie

Pomocné funkcie:

* int ZvolMaticu() –
* void Ukonci(MATICA\*\* m1, MATICA\*\* m) – Vymaže matice z pamäte a ukončí program.
  + MATICA\*\* m1 – adresa smerníka prvej matice
  + MATICA\*\* m2 – adresa smerníka druhej matice
* MATICA\* UrobNasobokKonstantou(MATICA\* m) – Vypýta si od užívateľa konštantu a potom ňou vynásobí zvolenú maticu
  + MATICA\* m – adresa matice ktorá sa bude násobiť
* void UlozMaticu(MATICA\* zdroj, MATICA\*\* m1, MATICA\*\* m2) – Vypýta si od užívateľa index cieľovej matice a uloží do nej adresu zdrojovej matice.
  + MATICA\* zdroj – adresa matice ktorá sa má uložiť
  + MATICA\*\* m1 – adresa smerníka prvej matice do ktorej sa môže hodnota uložiť
  + MATICA\*\* m2 – adresa smerníka druhej matice do ktorej sa môže hodnota uložiť
* void VypisMenu() – Vypíše zoznam funkcií, ktoré sa môžu vykonať.
* void VypisMatice(MATICA\* m1, MATICA\* m) – Vypíše obe matice na obrazovku.
  + MATICA\* m1 – adresa prvej matice
  + MATICA\* m2 – adresa druhej matice
* MATICA\* VytvorJednotkovu() – Vypýta si rozmer matice od užívateľa, na jeho základe vytvorí jednotkovú maticu a vráti jej adresu.