

Spital – Meniu administrativ

Gherasim Daniel Adrian

Grupa 3114 A

Facultatea de Inginerie Electrică și Știința Calculatoarelor, USV

Proiect de semestru realizat la disciplina PCLP II

Abstract

In cadrul acestui proiect am implementat in limbajul de programare C un meniu de tip administrativ ce poate fi folosit in cadrul unui spital.

Aplicatia este destinata atat managerului cat si persoanelor fizice. Sectiunea destinata persoanelor fizice este nesecurizata si contine urmatoarele posibilitati:

Un utilizator al aplicatiei poate primi informatii despre cele mai raspundite boli, are acces sa vada medicii structurati pe sectii, poate programa o consultatie sau poate programa o vizita.

Drept manager, o sectiune securizata, se poate vedea o statistica generala a medicilor, a internarilor, a vizitelor dar si a pacientilor externati. Pe langa aceste optiuni, mai are acces si la adaugarea / concedierea / cautarea unui cadru medical.

Keywords: meniu, manager, utilizator, spital.

Spital – Meniu administrativ

Programul a fost implementat pentru a facilita administrarea unui spital. Acesta usureaza foarte mult fluxul de informatii, medicul avand multe detalii despre pacient din mediul online, nemaifiind necesara completarea de documente la momentul consultatiei. Pe langa acest lucru, managerul ar avea o statistica clara cu tot ceea ce exista la momentul respectiv in spital, dar in acelasi timp poate adauga / concedia un cadru medical.

Proiectarea soluției

Pentru a se solutiona aceasta problema, s-a utilizat conceptul de lista inlantuita pentru a putea salva informatiile aferente medicilor, dar si a pacientilor. Pe langa acest concep, s-au folosit structuri alocate dinamic pentru a se retine informatiile referitoare la vizitatori. In aceste structuri au fost retinute: codul vizitei, numele pacientului vizitat, numele vizitatorului, ziua, luna si anul.

Aceasta este implementarea functiei de afisare a medicilor:

```

char *toString(LISTA l, char *s )
{
    system("cls");
    sprintf(s, "\n\tStatistica medicilor este: \n");
    assert(l != NULL);

    if(is_empty_l(l))
        strcat(s, "Nu exista medici inregistrati");
    else
    {
        char zona[l->nr*150];
        int max=0, nmax=0, umax=0, pmax=0;
        MEDIC w;
        for(w = l->primul; w != NULL ; w = w->urm)
        {
            if(max<strlen(w->sectie))
                max=strlen(w->sectie);

            if(nmax<strlen(w->numepr))
                nmax=strlen(w->numepr);

            if(pmax<strlen(w->speci))
                pmax=strlen(w->speci);

            if(w->cod>9)
                umax=1;
        }
        for(w = l->primul; w != NULL ; w = w->urm)
        {
            sprintf(zona, "NR CRT: %d", w->cod);
            strcat(s, zona);
            if(umax==1 && w->cod<10)
            {
                sprintf(zona, " ");
                strcat(s, zona);
            }
            sprintf(zona, " - SECTIE: %s", w->sectie);
            strcat(s, zona);
            if(strlen(w->sectie) != max)
            {
                int d= max-strlen(w->sectie);
                for(int j=0; j<d; j++)
                {

```

```

        sprintf(zona, " ");
        strcat(s, zona);
    }
}
sprintf(zona, " - NUME: %s", w->numepr);
strcat(s, zona);
if(strlen(w->numepr) != nmax)
{
    int d= nmax-strlen(w->numepr);
    for(int j=0; j<d; j++)
    {
        sprintf(zona, " ");
        strcat(s, zona);
    }
}
sprintf(zona, " - SPEC: %s", w->speci);
strcat(s, zona);
if(strlen(w->speci) != pmax)
{
    int d= pmax-strlen(w->speci);
    for(int j=0; j<d; j++)
    {
        sprintf(zona, " ");
        strcat(s, zona);
    }
}
sprintf(zona, " - MEDIC %s", w->stare);
strcat(s, zona);
}
return s;
}

```

Aceasta functie parcurge initial toata lista pentru a putea stabili care este dimensiunea fiecarui sir de caractere, iar apoi preia fiecare informatie din lista, o prelucreaza iar apoi o transpune ordonat intr-un sir de caractere care va fi apoi returnat.

Considerații de implementare

Acest proiect utilizeaza liste simplu inlantuite si structuri. Acesta ruleaza pe consola si este de tip meniu.

Pentru functionarea acestui proiect, s-au folosit urmatoarele functii:

Functii.h

void meniu(); - afiseaza meniul initial
void info(); - afiseaza datele despre autor
void meniu2(); - afiseaza meniul pentru manager
void meniu1(); - afiseaza meniul pentru persoanele fizice
void meniuaf(); - permite utilizatorului sa aleaga una din optiuni
void meniuaf2(); - permite managerului sa aleaga una din optiuni
int verificare(); - verifica daca datele de logare sunt corecte – returneaza 1 daca verificarea este valida / 0 in rest

void boli(); - permite utilizatorului sa aleaga una din optiunile din meniu
void afboli(); - afiseaza meniul cu bolile
void varicela(); - afiseaza informatii despre varicela
void rujeola(); - afiseaza meniul varicela
void rubeola(); - afiseaza meniul rubeola
void oreion(); - afiseaza meniul oreion
void manapicior(); - afiseaza meniul manapicior
void varicelace(); - permite utilizatorului sa aleaga una din optiunile din meniu
void varit1(); - afiseaza continutul primei optiuni
void varit2(); - afiseaza continutul celei de-a doua optiune
void varit3(); - afiseaza continutul celei de-a treia optiune
void varit4(); - afiseaza continutul celei de-a patra optiune
void rujeolace(); - permite utilizatorului sa aleaga una din optiunile din meniu
void rujt1(); - afiseaza continutul primei optiuni
void rujt2(); - afiseaza continutul celei de-a doua optiune
void rujt3(); - afiseaza continutul celei de-a treia optiune
void rujt4(); - afiseaza continutul celei de-a patra optiune
void rubeolace(); - permite utilizatorului sa aleaga una din optiunile din meniu
void rubt1(); - afiseaza continutul primei optiuni
void rubt2(); - afiseaza continutul celei de-a doua optiune
void rubt3(); - afiseaza continutul celei de-a treia optiune
void rubt4(); - afiseaza continutul celei de-a patra optiune
void oreionce(); - permite utilizatorului sa aleaga una din optiunile din meniu
void oreit1(); - afiseaza continutul primei optiuni
void oreit2(); - afiseaza continutul celei de-a doua optiune

void oreit3(); - afiseaza continutul celei de-a treia optiune
void oreit4(); - afiseaza continutul celei de-a patra optiune
void manapiciorce(); - permite utilizatorului sa aleaga una din optiunile din meniu
void manat1(); - afiseaza continutul primei optiuni
void manat2(); - afiseaza continutul celei de-a doua optiune
void manat3(); - afiseaza continutul celei de-a treia optiune
void manat4(); - afiseaza continutul celei de-a patra optiune

Lista.h

LISTA inserare (LISTA l , int nr, char *sect,char *numep, char *spec, char *sta);
- Alocă memorie pentru o structură și o inițializează cu ajutorul a altei funcții
- Returnează lista
LISTA newl(); - creează o nouă listă – returnează listă
int nr_elemente(LISTA l); - returnează număr de elemente
bool is_empty_l(LISTA l); - returnează '1' dacă listă este goală, 0 în rest
bool is_full_l(LISTA l); - returnează '1' dacă listă este plină, 0 în rest
void destroy_l(LISTA l); - dealocă memoria
LISTA adaugamedic (LISTA l); - funcție destinată adăugării de cadre medicale – returnează o listă
char *toString(LISTA l,char *s); - returnează un string ce conține toate cadrele medicale structurate
LISTA citiref (); - Această funcție citește datele din fișier și le introduce în listă – returnează listă
void cauta(LISTA l, int k); - caută în listă cadrul medical cu numărul 'k'. Dacă îl găsește, îl afișează
LISTA concediaza (LISTA l, int k); - caută în listă cadrul medical cu numărul 'k'. Dacă îl găsește, îl șterge și dealocă memoria

LISTA sterge_primul(LISTA l); - șterge primul element din listă
LISTA sterge_ultimul(LISTA l); - șterge ultimul element din listă
void salvaref (LISTA l); - salvează listă în fișierul test aferent
void salvaremedic (LISTA l); - salvează numele medicului într-un fișier
char* sectii(); - oferă posibilitatea de a alege o secție din meniu – returnează numele secției
char* staremed(); - oferă posibilitatea de a alege o stare a unui medic – returnează starea
void sectiihud(LISTA l);
void medhud(LISTA l,char *buf);

Manualul de utilizare al aplicației

In momentul executarii programului se va fi afisa urmatorul meniu:

```
#####MENIU#####
1. Optiune disponibila persoanelor fizice
2. Optiunea disponibila pentru manager
3. Informatii
4. Terminare program
#####
Optiunea aleasa este:
```

Acestea sunt optiunile disponibile.

In momentul selectarii celei de-a treia optiuni se va afisa:

```
#####MENIU#####
1. Optiune disponibila persoanelor fizice
2. Optiunea disponibila pentru manager
3. Informatii
4. Terminare program
#####
Optiunea aleasa este:
Acest program a fost realizat de catre Gherasim Daniel Adrian. Grupa 3114A. Specializarea: Calculatoare
```

Daca se selecteaza a patra optiune, programul se va incheia.

```
#####MENIU#####
1. Optiune disponibila persoanelor fizice
2. Optiunea disponibila pentru manager
3. Informatii
4. Terminare program
#####
Optiunea aleasa este:Programul s-a incheiat

Process returned 0 (0x0)   execution time : 51.396 s
Press any key to continue.
```

In cazul in care se va selecta prima optiune, vom putea vizualiza disponibilitatile:

```
1. Informatii boli
2. Sectii
3. Statistica pacienti
4. Programeaza o consultatie
5. Programeaza o externare
6. Programeaza o vizita
7. Regulii pentru vizitatori
8. Revina la meniul anterior
Optiunea aleasa este:
```

Alegand prima optiune putem vizualiza detalii despre cele mai intalnite boli ale copilului:


```
Cele mai frecvente probleme medicale ale copilului:
```

```
Bolile Copilariei
```

1. Varicela
2. Rujeola
3. Rubeola
4. Parotidita (oreionul)
5. Boala mana-picior-gura
6. Meniu anterior

```
Optiunea aleasa este:
```

Revenind la meniul anterior, daca se selecteaza a doua optiune, se pot vizualiza cadrele medicale organizate pe sectii:

```
SECTII SPITAL
```

1. Sectia ATI
 2. Sectia CHIRURGIE
 3. Sectia STOMATOLOGIE
 4. Sectia MEDICINA INTERNA
 5. Sectia NEUROLOGIE
 6. Sectia ONCOLOGIE
 7. Meniu anterior
- ```
Optiunea aleasa este:
```

Verificand sectiile, se pot observa medicii

```
SECTIA ATI
```

```
NR CRT: 1 - SECTIE: ATI - NUME: default - SPEC: default - MEDIC PRIMAR
NR CRT: 2 - SECTIE: ATI - NUME: DSD - SPEC: DSD - MEDIC PRIMAR
NR CRT: 3 - SECTIE: ATI - NUME: testare - SPEC: testare - MEDIC SECUNDAR
```

```
SECTIA ONCOLOGIE
```

```
Nu exista medici inregistrati
```

In cazul in care se doreste sa se faca o programare:

```
Numele este: tastare
CNP: 1234567890123
Data curenta este: 10-05-2020
Puteti face o programare pana pe data de: 17-05-2020
Ziua in care vrei sa fie programarea este: 16
Cauza solicitarii este: grav bolnav
Card de sanatate?[DA/NU]: da
Numele medicului la care doresti sa te programezi este: default
Programarea a fost realizata!
```

In cazul in care un medic doreste sa realizeze o externare, se acceseaza optiunea 5 din meniu iar pacientul va fi externat.

```
Numarul pacientului pe care vrei sa-l: 2
Numele medicului care aproba externarea este: default
Pacientul a fost externat
```

Pentru a realiza o vizita, prima data trebuie citit regulamentul de vizite. In cazul in care nu se citește acesta înainte de a accesa optiunea de a programa o vizita, programul va afisa automat regulamentul. In caz contrar, se va trece peste acest pas.

```
Pentru a putea face o vizita mai intai trebuie sa citesti regulamentul
```

```
Numele pacientului pe care vrei sa-l vizitezi este: testare program
Numele tau este: Catalin teste
Data curenta este: 10-05-2020
Puteti programa o vizita pana pe data de: 17-05-2020
Ziua in care vrei sa faci vizita este: 15
Vizita a fost inregistrata cu succes
```

In cazul in care se va alege optiunea pentru manager se va solicita autentificarea:

```
#####AUTENTIFICARE#####
Username:admin
Password:
```

In cazul in care datele sunt corecte, se va accesa meniul corespunzator

```
1. Statistica cadre medicale
2. Adauga un cadru medical
3. Concediaza un cadru medical
4. Cauta un medic
5. Numar internari/ext/vizite
6. Statistica internari
7. Statistica externari
8. Statistica vizite
9. Revina la meniul anterior
Optiunea aleasa este:
```

Tasta 1 afiseaza o statistica a tuturor cadrelor medicale:

```
Statistica medicilor este:
NR CRT: 1 - SECTIE: ATI - NUME: default - SPEC: default - MEDIC PRIMAR
NR CRT: 2 - SECTIE: ATI - NUME: DSD - SPEC: DSD - MEDIC PRIMAR
NR CRT: 3 - SECTIE: CHIRURGIE - NUME: test2 - SPEC: test3 - MEDIC SECUNDAR
NR CRT: 4 - SECTIE: STOMATOLOGIE - NUME: test - SPEC: test5 - MEDIC SECUNDAR
NR CRT: 5 - SECTIE: MEDICINA INTERNA - NUME: test6 - SPEC: test5 - MEDIC SECUNDAR
NR CRT: 6 - SECTIE: NEUROLOGIE - NUME: default91 - SPEC: test2 - MEDIC PRIMAR
NR CRT: 7 - SECTIE: ATI - NUME: testare - SPEC: testare - MEDIC SECUNDAR
```

In cazul in care se cauta un medic, afisarea este urmatoarea:

```
Numarul medicului pe care il cauti este:
Medicul cu numarul 1 este:
Nume: default
Sectie: ATI
Specializare: default
Caracteristica: MEDIC PRIMAR
```

Tasta 5 afiseaza numarul de internari / externari / vizite:

```

Numar internari: 2
Numar externari: 3
Numar vizite : 5

```

Tasta 6 confera posibilitatea managerului de a vedea o statistica generala a internarilor:

```
Statistica pacientilor este:
NR CRT: 1 - CNP: 1234567890123 - NUME: testare program - DATA: 16/05/2020 - MOTIV: bolnav grav - CARD: DA - Medic: DEFAULT
NR CRT: 2 - CNP: 1234567890123 - NUME: tastare - DATA: 16/05/2020 - MOTIV: grav bolnav - CARD: DA - Medic: DEFAULT
```

Iar tastele 7 si 8 sunt de asemenea sunt statistici.

#### Referințe bibliografice

[https://www.tutorialspoint.com/c\\_standard\\_library/index.htm](https://www.tutorialspoint.com/c_standard_library/index.htm)