

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

typedef struct aluno_ {
    char nome[50];
    int n_usp;
    float p1;
    float p2;
    float media;
} ALUNO;

ALUNO** aloca_aluno(){
    ALUNO** aluno = (ALUNO**) malloc(100*sizeof(ALUNO*));
    return aluno;
}

ALUNO* insere_aluno(){
    ALUNO* aluno = (ALUNO*) malloc(sizeof(ALUNO));

    printf("Digite o nome:");
    fgets(aluno->nome,50,stdin);
    aluno->nome[strlen(aluno->nome)-1] = '\0';

    printf("Digite o numero USP:");
    scanf("%d", &aluno->n_usp);

    printf("Digite a p1:");
    scanf("%f", &aluno->p1);

    printf("Digite a p2:");
    scanf("%f", &aluno->p2);

    aluno->media = (aluno->p1 + aluno->p2)/2;

    return aluno;
}

void exhibe_aluno(ALUNO** aluno,int n_usp, int index){

    for(int i = 0; i < index; i++){
        if(aluno[i]->n_usp == n_usp){

```

```

        printf("Nome:%s\n", aluno[i]->nome);
        printf("Numero USP:%d\n", aluno[i]->n_usp);
        printf("P1:%.2f\n", aluno[i]->p1);
        printf("P2:%.2f\n", aluno[i]->p2);
        printf("Media Final:%.2f\n", aluno[i]->media);

        return;
    }
}

printf("Aluno nao encontrado\n");
}

void apaga_aluno(ALUNO** aluno){
    free(*aluno);
    *aluno = NULL;
}

int main(){

    ALUNO** aluno = aloca_aluno();

    int n, n_usp;
    int index = 0;

    while(1){

        printf("/-----MENU-----\\\n");
        printf("|1: Inserir aluno      |\n");
        printf("|2: Exibir aluno       |\n");
        printf("|3: Saida              |\n");
        printf("\\\n-----/\n");

        printf("Digite a sua opcao:");

        scanf("%d", &n);

        setbuf(stdin, NULL);

        if(n==1){
            aluno[index] = insere_aluno();
            index++;
        }
    }
}

```

```
        if(n==2){
            printf("Digite o Numero USP do aluno:");
            scanf("%d", &n_usp);
            exhibe_aluno(aluno,n_usp,index);
        }

        if(n==3) break;

        printf("\n");

    }

    printf("Apagando Dados...\n");

    for(int i = 0; i < index; i++){
        apaga_aluno(&aluno[i]);
    }

    free(aluno);
    aluno = NULL;

    return 0;
}
```