

O trabalho consistiu em criar um computador em C. Basicamente o computador é dividido em memórias, secundaria, ram, cache, registradores, e em C, tentamos simular a mesma coisa, fazendo o processamento de arquivos txt para simular operacoes:

```
1 74 1 25 0 50 2
0 87 0 39 1 76 3
1 16 3 43 1 15 1
0 30 2 44 0 62 2
1 44 3 53 3 16 0
```

Essa e a versao final do miniPC, em que há a criacao de caches e RAM:

```
alocaMemorias();
montarRAM();
montarCacheVazio(tamanhoCache1, cache1);
montarCacheVazio(tamanhoCache2, cache2);
montarCacheVazio(tamanhoCache3, cache3);
```

As principais informações para se manter nas memórias de rapido acesso depende de parâmetros como último acesso ou mais acessos:

```

int algoritmoSubstituicaoLRU(int conjuntoCache, int tamanhoCache){
    time_t posicaoMenosUsada;
    time(&posicaoMenosUsada);

    int posicaoCacheAux = 0;
    int contadorCache1 = 0;
    int contadorCache2 = 0;
    int contadorCache3 = 0;

    if(tamanhoCache == tamanhoCache1){
        for(int i = conjuntoCache * 4; i < conjuntoCache * 4 + 4; i++){
            if(cache1[i].segundos <= posicaoMenosUsada){
                posicaoMenosUsada = cache1[i].segundos;
                posicaoCacheAux = conjuntoCache * 4 + contadorCache1;
            }
            contadorCache1++;
        }
    }else if(tamanhoCache == tamanhoCache2){
        for(int i = conjuntoCache * 4; i < conjuntoCache * 4 + 4; i++){
            if(cache2[i].segundos <= posicaoMenosUsada){
                posicaoMenosUsada = cache2[i].segundos;
                posicaoCacheAux = conjuntoCache * 4 + contadorCache2;
            }
            contadorCache2++;
        }
    }else if(tamanhoCache == tamanhoCache3){
        for(int i = conjuntoCache * 4; i < conjuntoCache * 4 + 4; i++){
            if(cache3[i].segundos <= posicaoMenosUsada){
                posicaoMenosUsada = cache3[i].segundos;
                posicaoCacheAux = conjuntoCache * 4 + contadorCache3;
            }
            contadorCache3++;
        }
    }else if(tamanhoCache == tamanhoRAM){
        for(int i = 0; i <= tamanhoRAM; i++){
            if(RAM[i].segundos <= posicaoMenosUsada){
                posicaoMenosUsada = RAM[i].segundos;
                posicaoCacheAux = i;
            }
        }
    }

    return posicaoCacheAux;
}

```

Alem de mensurar o cache hit e cache miss das aplicacoes, onde o mini computador fica fazendo cálculos aleatórios com os dados passados e mostrando se as informações foram tiradas da RAM, registradores ou cache.

