

Основи рачунарских система 3 / Оперативни системи

Семинарски рад - Група 13

У неком од програмских језика реализовати симулатор оперативног система који ради на једном језгру у виду конзолне апликације. Студентима се препоручује да користе Јава или С (Ц) програмски језик. Овом приликом је потребно користити Git или неки сличан алат.

Основни елементи су следећи:

1. Процедура за подизање оперативног система – booting.
2. Реализација распоређивача процеса – Реализовати кружни алгоритам.
3. Реализација техника управљања меморијом – фиксно партиционисање меморије.
4. Реализација фајл система – дрволико (као што је иначе у пракси код већине ОС).
5. *Реализација интеракције са улазно/излазним уређајима – нека улаз увијек буде тастатура, а излаз монитор. Ова интеракција се може реализовати путем бафера и технике pooling или система догађаја. Овај дио задатка је опциони. Уколико студентима овај захтјев дјелује претежак, могу користити уграђене функције за читање и писање по конзоли.
6. Реализовати креирање, читање и брисање датотека – непрекидно алоцирање меморије.
7. Осмислити команде оперативног система које треба да садрже најосновније ствари за које ће омогућити покретање већег броја процеса. Студенти имају слободу да наведене команде дизајнирају како желе или да искористе неки linux/windows командни језике.

Основне команде су:

- a) Команда за листање датотека и поддиректоријума у актуелном директоријуму
 - b) Команда за промјену актуелног директоријума
 - c) Команда за креирање, брисање и преименовање директоријума
 - d) Команда за покретање извршне датотеке и слање у позадину како би могао да се покрене нови процес – подразумијева се моментално слање у позадину. То значи да се улазни параметри увијек прослеђују преко текстуалне датотеке, те се резултат извршавања такође исписује у неку излазну текстуалну датотеку. Позив може да буде на следећи наћин:

 <назив команде> <назив извршне датотеке> <назив датотеке са резултатима>
 - e) Листање актуелних процеса – за сваки процес приказати неке стандардне информације попут: пид, имена програма, тренутно стање процеса (према животном циклусу), тренутно заузеће меморије и слично.
 - f) Прекид неког процеса
 - g) Блокирање и одблокирање неког процеса
 - h) Гашење симулатора
8. Креирати једноставан асемблер. Овом приликом је потребно креирати највише десетак основних наредби асемблера (пребацивање из/у меморију са регистра, сабирање, одузимање, множење, дијељење, наредба скока и слично). Овом приликом је потребно направити симулацију регистара. Приликом учитавања асемблерског кода (текстуалне датотеке са екстензијом .asm), потребно је генерисати одговарајући машински код, те га потом извршити. Омогућити приказ меморије и регистара.