

1. (20) Равноправно решење критичне секције помоћу *Ticket* алгоритма. *Fine grain* решење реализовано помоћу *Fetch and Add*.

2. (20) Монитор за Interval timer са приоритетним чекањем.

3. (20) Плење људождера једе заједничку вечеру из казана који може да прими М порција куваних мисионара (*The dining savages problem*). Када људождер пожели да руча онда се он сам послужи из заједничког казана, уколико казан није празан. Уколико је казан празан људождер буди куvara и сачека док кувар не напуни казан. Није дозвољено будити куvara уколико се налази бар мало хране у казану. Користећи условне критичне регионе написати програм који симулира понашање људождера и куvara.

4. (20) Користећи C-Linda написати програм који решава проблем путовања лифтом. Путник позива лифт са произвољног спрата. Када лифт стигне на неки спрат сви путници који су изразили жељу да сиђу на том спрату обавезно изађу. Након изласка путника сви путници који су чекали на улазак уђу у лифт и кажу на који спрат желе да пређу. Тек када се сви изјасне лифт прелази даље. Није потребно оптимизовати пут лифта и путника.

5. (20) Написати програм на језику Java који обезбеђује пословање унутар банке. Банка поседује клијенте и централни сервер. Програм треба да обезбеди могућност уплате и исплате новца као и увида у текуће стање. Потребно је обезбедити максималну конкурентност међу клијентским **програмима**.

*Испит траје 3 сата.*

1. (20) Равноправно решење критичне секције помоћу *Ticket* алгоритма. *Fine grain* решење реализовано помоћу *Fetch and Add*.

2. (20) Монитор за Interval timer са приоритетним чекањем.

3. (20) Плење људождера једе заједничку вечеру из казана који може да прими М порција куваних мисионара (*The dining savages problem*). Када људождер пожели да руча онда се он сам послужи из заједничког казана, уколико казан није празан. Уколико је казан празан људождер буди куvara и сачека док кувар не напуни казан. Није дозвољено будити куvara уколико се налази бар мало хране у казану. Користећи условне критичне регионе написати програм који симулира понашање људождера и куvara.

4. (20) Користећи C-Linda написати програм који решава проблем путовања лифтом. Путник позива лифт са произвољног спрата. Када лифт стигне на неки спрат сви путници који су изразили жељу да сиђу на том спрату обавезно изађу. Након изласка путника сви путници који су чекали на улазак уђу у лифт и кажу на који спрат желе да пређу. Тек када се сви изјасне лифт прелази даље. Није потребно оптимизовати пут лифта и путника.

5. (20) Написати програм на језику Java који обезбеђује пословање унутар банке. Банка поседује клијенте и централни сервер. Програм треба да обезбеди могућност уплате и исплате новца као и увида у текуће стање. Потребно је обезбедити максималну конкурентност међу клијентским **програмима**.

*Испит траје 3 сата.*