

СИСТЕМИ ЗА ПРЕСМЕТКИ СО ВИСОКИ ПЕРФОРМАНСИ

26.11.2022

1. **(60 поени)** Да се напише MPI програма која ќе симулира наградна игра. Главниот процес е организатор на играта, додека останатите $p-1$ процеси се натпреваруваат во истата. Пред почетокот на играта кај главниот процес се генерираат $n \cdot (p-1)$ тикети, каде n е случаен број. Секој тикет е различен и е составен од 7 неповторувачки броеви (во опсегот [1-21]). Секој процес треба да добие еднаков број на тикети (n) од генерираните тикети кај главниот процес. Откако секој од процесите ќе ги добие своите тикети, играта започнува. Во еден циклус на играта, главниот процес извлекува 7 неповторувачки броеви, а останатите процеси проверуваат дали го имаат тикетот кој во таа итерација го генерирал главниот процес. Доколку се најде победнички тикет, играта завршува. При завршување на играта, главниот процес известува кој од останатите процеси е победникот, додека победничкиот процес го печати добитниот тикет. Доколку во циклусот не се најде добитнички тикет, се отпочнува нов циклус и постапката се повторува се додека не се добие победник.
2. **(40 поени)** Да се напише MPI програма која ќе ги бара целобројните решенија на системот равенки:

$$\begin{cases} 2x + y + 2z = 6 \\ (x + 1)^2 + (y - 10)^2 + 2z = 10 \end{cases}$$

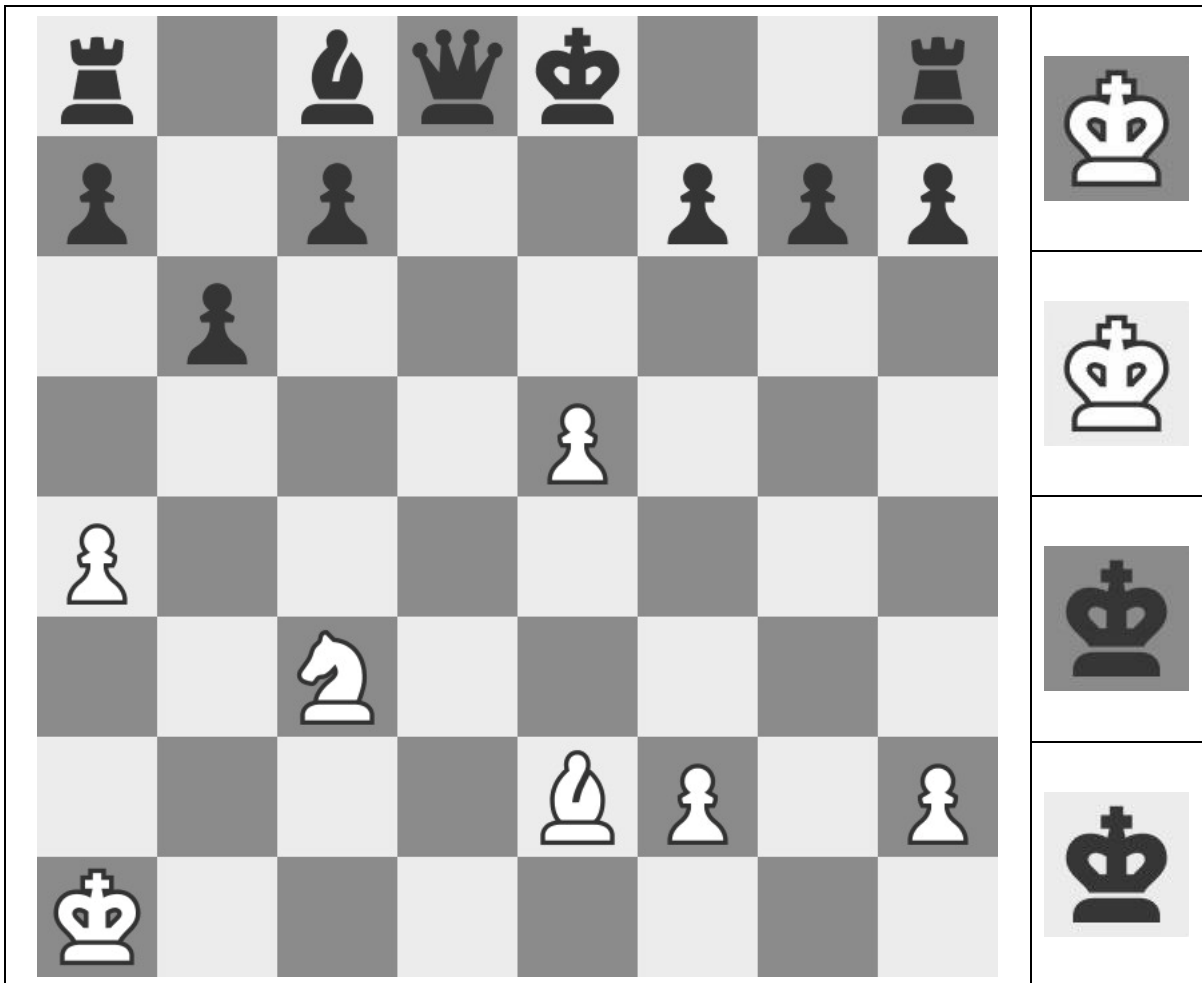
Работата да се распредели рамномерно помеѓу процесите. Да се испробаат сите можни комбинации на променливите x, y, z во опсегот [-32768, 32767].

Најдените решенија да се вратат до главниот процес кој треба да ги прикаже на екран.

СИСТЕМИ ЗА ПРЕСМЕТКИ СО ВИСОКИ ПЕРФОРМАНСИ

23.11.2021

- (60 поени)** Да се напише MPI програма која ќе овозможи пронаоѓање на фигурата крал во слика од шаховска табла. Сликата од шаховската табла претставува матрица со димензи $n \times n$, при што на почеток димензијата на сликата не е позната (се внесува од тастатура). За споредбата на располагање се уште четири помошни слики (кои го овозможуваат пребарувањето на фигурата, т.е. даваат како треба да изгледа фигурата). Четирите помошни слики се квадратни матрици со димензии $m \times m$, при што димензиите се познати преку вредноста n .



Споредбата се врши на тој начин што содржината на секое поле од шаховската табла треба да се спореди со сите четири помошни слики за еднаквост. Доколку се најде разлика во некој дел од таблата со помошните слики споредбата се смета за неуспешна.

Секој процес треба да го врати бројот на решенија кои ги има пронајдено како и позицијата на шаховската табла на која е пронајдено решението.

Главниот процес на крај треба да провери дали бројот на решенија е точно два и во тој случај треба да ги испечати информациите за позициите на кои биле пронајдени решенијата. Ако бројот на решенија се разликува од два, главниот процес треба да испечати порака за грешка, по што примените информации не се печатат поради невалидност на фигурите.

Напомена: Содржината на матриците можете да ја генерирате произволно во опсег на вредности [0-255]. Генерирањето се врши во главниот процес.

Матриците од примерот ви се дадени во дополнително прашање.

2. (40 поени) Да се напише MPI програма која ќе ги бара целобројните решенија на равенката:

$$x^4 * z + 3 * y^2 * z - 4 * z = 20$$

Работата да се распредели рамномерно помеѓу повеќе процеси. Да се испробаат сите можни комбинации на променливите x , y , z во опсегот $[-32768, 32767]$.

Најдените решенија да се вратат до главниот процес кој треба да ги прикаже на екран.

ВТОР КОЛОКВИУМ ПО

СИСТЕМИ ЗА ПРЕСМЕТКИ СО ВИСОКИ ПЕРФОРМАНСИ

20.01.2023

1. **(50 поени)** Да се напише програма во која од тастатура се вчитува матрица од цели броеви. Во матрицата треба да се пронајдат две колони кои ги имаат двете најголеми суми на елементи на ниво на целата матрица. Дополнително, елементите во двете колони треба да си ги заменат местата, а потоа во тие две колони елементите треба да се пресликаат огледално во однос на хоризонталата. Барањето на двете најголеми суми, замената на колоните и пресликувањето треба да бидат паралелизирани.

Барањата да се реализираат со помош на OpenMP библиотеката.

Пример:

За дадена матрица:

$$\begin{bmatrix} 1 & 5 & 1 & 1 & 2 \\ 2 & 4 & 9 & 4 & 4 \\ 5 & 7 & 2 & 3 & 6 \\ 6 & 7 & 1 & 2 & 9 \end{bmatrix}$$
, сумите по колоните се соодветно: 14, 23, 13, 10, 21, што значи

дека треба да се заменат колоните со суми 23 и 21 (бидејќи тоа се двете најголеми суми), па по замената матрицата изгледа:

$$\begin{bmatrix} 1 & 2 & 1 & 1 & 5 \\ 2 & 4 & 9 & 4 & 4 \\ 5 & 6 & 2 & 3 & 7 \\ 6 & 9 & 1 & 2 & 7 \end{bmatrix}$$
. Откако ќе се заменат колоните, треба да се направи и огледално

пресликување на двете во однос на хоризонталата (т.е. во првата колона треба 2 да се замени со 9 и 4 да се замени со 6, додека во втората колона треба 5 да се замени со 7 и 4 да се замени со 7), па финалната матрица ќе биде:

$$\begin{bmatrix} 1 & 9 & 1 & 1 & 7 \\ 2 & 6 & 9 & 4 & 7 \\ 5 & 4 & 2 & 3 & 4 \\ 6 & 2 & 1 & 2 & 5 \end{bmatrix}$$
.

2. **(50 поени)** Да се напише pthreads програма во која од тастатура се вчитува низа од цели броеви. Програмата треба да ја најде најдолгата подниза за која важи правилото: *секој број е еднаков на збирот од претходните два броја*. Пребарувањето да се раздели рамномерно помеѓу повеќе нитки, при што на крајот програмата треба да го испечати редниот број на елементот од каде започнува бараната подниза.

ВТОР КОЛОКВИУМ ПО

СИСТЕМИ ЗА ПРЕСМЕТКИ СО ВИСОКИ ПЕРФОРМАНСИ

18.01.2022

1. Да се напише програма која ќе дозволи внесување на димензиите на една матрица (од тастатура), а потоа (исто така од тастатура) ќе овозможи да се внесуваат елементите на матрицата (реални броеви). Исто така, по внесувањето на елементите на матрицата, од тастатура треба да се внесе реален број X .

Потоа со помош на нитки (користејќи ја `pthread` библиотеката) програмата треба да ја измине матрицата по редици и да види во која колона на секоја редица, разликата на два соседни елементи од редицата е поголема од вредноста на X . Доколку се најдат вакви два соседни елементи во редицата, се печати редниот број на редицата, редниот број на колоната на десниот елемент (од двата). Разликата на два соседни броеви се пресметува по апсолутна вредност. Доколку во редицата има повеќе вакви соседни броеви, само првиот пат се печати редицата, а после тоа се печати само колоната.

На влез:

$A = \begin{matrix} 1.7 & 2.0 & 10.1 & 0.5 \\ 7.3 & 2.2 & 3.4 & 1.01 \\ 6.0 & 5.1 & 5.2 & 4.1 \end{matrix}$

$X = 4.5$

**На излез: редица 0, колона 2 и 3
редица 1, колона 1**

2. Да се напише OpenMP програма која ќе ги бара целобројните решенија на равенката:

$$6x^4 + (2xyz)^3 - 3xy^2 + 15 = 0$$

Работата да се распредели рамномерно помеѓу повеќе нитки. Да се испробаат сите можни комбинации на променливите x , y и z во опсегот $[-32768, 32767]$.

На крај, во главната нитка да се прикажат сите решенија на равенката.

СИСТЕМИ ЗА ПРЕСМЕТКИ СО ВИСОКИ ПЕРФОРМАНСИ

15.02.2023

1. **(50 поени)** Да се напише MPI програма во која од тастатура се вчитува низа од цели броеви и се проверува дали низата е растечка или опаѓачка. Доколку низата не е ниту растечка, ниту опаѓачка, процесите не треба да продолжат со работа. Но, доколку се утврди дека низата е или растечка или опаѓачка, процесите треба истата да ја конвертираат во строго растечка/опаѓачка (односно, да ги отстранат сите елементи кои се еднакви на нивниот претходник).

Во процесот со $id=0$ треба да се вчита низата од тастатура, и на крај да се испечати низата. Работата да се распредели рамномерно помеѓу повеќе процеси.

2. **(50 поени)** Да се напише pthread програма која по случаен избор ќе генерира 200 случајни комбинации за вредности на три променливи x , y и z . Секоја (x, y, z) комбинација од 200-те генерирани комбинации треба да биде уникатна. Генерирањето рамномерно да се распредели помеѓу сите активни pthread нитки.

Следно, да се напише OpenMP програма која ќе ги бара целобројните решенија на равенката:

$$8x^6 + (2xyz)^3 - 7xy + 5x = -15$$

Во OpenMP програмата да се испроба секоја од 200-те случајно генерирани комбинации на променливите x , y и z . Пребарувањето на решенија рамномерно да се распредели помеѓу сите активни openmp нитки.

Најдените решенија, ако постојат, на крај треба да се отпечатат во екран во нултиот процес.

ИСПИТ ПО

СИСТЕМИ ЗА ПРЕСМЕТКИ СО ВИСОКИ ПЕРФОРМАНСИ

15.06.2023

1. **(50 поени)** Да се напише MPI програма во која од тастатура во процесот со $id = 0$ се вчитува низа од знаци. Програмата треба да провери дали внесената низа од знаци е палиндром. Палиндром претставува низа од знаци која се чита исто и од почеток и од крај. Пребарувањето да се раздели рамномерно помеѓу повеќе процеси, така што секој процес ќе го добива само својот дел од низата. На крај, во главниот процес треба да се прикаже дали низата претставува палиндром или не.
2. **(50 поени)** Да се напише pthread програма која по случаен избор ќе генерира 100 случајни комбинации за вредности на три променливи x , y и z . Секоја (x, y, z) комбинација од 100-те генерирани комбинации треба да биде уникатна. Генерирањето рамномерно да се распредели помеѓу сите активни pthread нитки.

Следно, да се напише OpenMP програма која ќе ги бара целобројните решенија на равенката:

$$8x^4 + 2(xz)^3 - 4xy + 8x = -15$$

Во OpenMP програмата да се испроба секоја од 100-те случајно генерирани комбинации на променливите x , y и z . Пребарувањето на решенија рамномерно да се распредели помеѓу сите активни openmp нитки.

Најдените решенија, ако постојат, на крај треба да се отпечатат во екран во нулатата нитка.

ИСПИТ ПО

СИСТЕМИ ЗА ПРЕСМЕТКИ СО ВИСОКИ ПЕРФОРМАНСИ

15.09.2023

1. **(40 поени)** Да се напише MPI програма која ќе ги бара целобројните решенија на равенката:

$$x^4 * z + 3 * y^2 * z - 4 * z = 20$$

Работата да се распредели рамномерно помеѓу повеќе процеси. Да се испробаат сите можни комбинации на променливите x, y, z во опсегот $[-32768, 32767]$.

Најдените решенија да се вратат до главниот процес кој треба да ги прикаже на екран.

2. **(60 поени)** Да се напише програма во која во нулта нитка ќе се прочита квадратна матрица со димензии $n \times n$, каде n исто така се внесува од тастатура. Програмата треба да провери дали внесената матрица е единечна матрица. Доколку матрицата е единечна, тогаш главната нитка треба да ја отпечати таа информација. Во спротивно, доколку матрицата не е единечна, тогаш секоја нитка треба својот дел од матрицата да го промени така што на крајот ќе се добие единечна матрица. Проверката на матрицата и промената на матрицата (доколку тоа е потребно) треба да бидат распределени рамномерно помеѓу нитките. Единечна матрица претставува матрица во која сите елементи се нули, освен елементите сместени по должина на главната дијагонала кои се единици.

За проверка на единечност на матрица да се користи `pthread`, а за промена на матрицата во единечна да се користи `openmp` библиотеката. Замена на библиотеките не е дозволена.