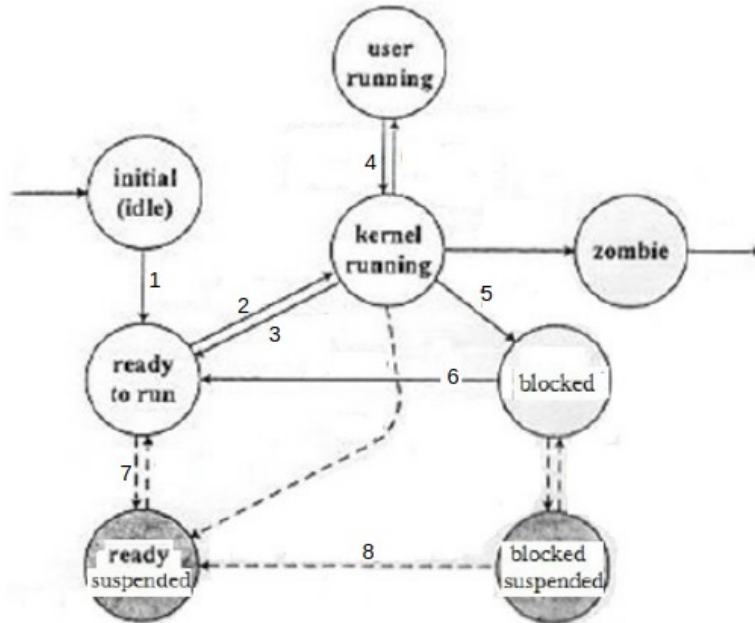


UFES – Departamento de Informática
1ª. Prova de Sistemas Operacionais – 2019/2 – Profª. Roberta L. Gomes

- 1) (2,5) Dado o diagrama de estados do Unix ilustrado abaixo, escreva em que situações (e por que) ocorre cada uma das transições enumeradas (1 a 8).



- 2) (2,5) Descreva de forma geral como funciona o Escalonamento Tradicional no UNIX. Diga quais são as principais limitações desse escalonamento.
- 3) (2,0) Em algumas implementações do UNIX, o kernel é não-preemptivo. O que isto significa? Quais as vantagens e desvantagens dessa abordagem?
- 4) (3,0) O código a seguir pode gerar a sequência de mensagens ilustradas no quadro “SAÍDA” exibido abaixo? Explique sua resposta construindo um grafo de precedência em que os nós do grafo são os *prints* executados pelo código (“S1”, “S2”, etc.).

Obs: Um grafo de precedência é um grafo direcionado em que a relação $(x) \rightarrow (y)$ indica que 'x' precede 'y', isto é, 'x' ocorre antes de 'y'. Para esta questão, os nós do grafo são “S1”, “S2”, etc.

```

int main()
{ int f1,f2,f3;
  printf("S1\n");
  if(!(f1=fork())) {
    execlp("prog5","prog5",NULL);
    printf("S9\n");
  }
  if(!(f2=fork())){
    execlp("prog7","prog7",NULL);
    wait(NULL);
  }
  printf("S2\n");
  if(!(f3=fork())){
    execlp("prog3","prog3",NULL);
  }
  printf("S4\n");
  waitpid(f1,NULL,NULL);
  printf("S6\n");
  wait(NULL);
  wait(NULL);
  printf("S8\n");
}

```

```

//prog3.c ...
int main()
{   printf("S3\n"); }

```

```

//prog5.c ...
int main()
{   printf("S5\n"); }

```

```

//prog7.c ...
int main()
{   printf("S7\n"); }

```

// SAÍDA

S1
S2
S4
S5
S6
S7
S3
S9
S8

Boa prova!!!