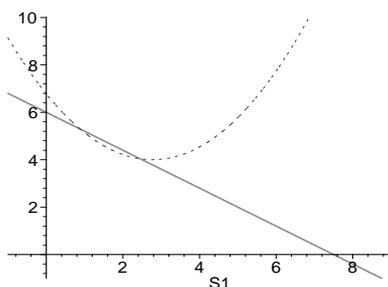


Решение

```
> R:=solve({seq(eq[i],i=1..6)},{S2,Xc,Yc,Ya,Xb,Yb});
> assign(R):
R:={Xb=4, Yb=-5/3, S2=6-4S1/5, Xc=-2, Ya=5/3, Yc=-5/3+3S1/5}
> Rc:=Xc^2+Yc^2:# Реакция шарнира
Минимум реакции
> simplify(minimize(sqrt(Rc),location)[2]);
{{S1=25/9},2}
Графики зависимостей усилий от натяжения S1
> plot([Rc,S2],S1=-1..9,-1..10,
>      thickness=[2,4],linestyle=[dash,solid]);
```



```
> S1:=solve(diff(Rc,S1),S1);
S1:=25/9
> s2=S2, xc=Xc, yc=Yc, ya=Ya, xb=Xb, yb=Yb;#Ответы
s2=34/9, xc=-2, yc=0, ya=5/3, xb=4, yb=-5/3
```

7.4. Ферма. Маплет

Рассмотрим решение задачи 11 о ферме на с. 32. Пронумеруем узлы и стержни. Опоры моделируем стержнями, неподвижную опору A — двумя стержнями 10 и 11, подвижную B — вертикальным стержнем 12. Все стержни фермы (включая опорные) условно представляем векторами, направление которых выбираем произвольно (рис. 400). Эти векторы не связаны с усилиями в стержнях, и

решение не зависит от выбора их направления. В программу вводятся концы стержней в виде векторов-строк N_{beg} , N_{end} . В программе используется метод вырезания узлов¹. Опорные стержни-векторы должны быть направлены от фермы к неподвижным узлам.

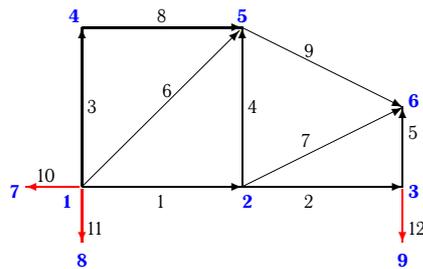


Рис. 400

Кроме того, опорные шарниры нумеруются последними. Последние условия существенные и невыполнение их часто является причиной ошибок ввода. Связано это с условным оператором $N_{end}[i] \leq N-K$, работающем в цикле по числу стержней. Усилия во всех стержнях, кроме опорных,

входят в систему дважды — в уравнение равновесия узла-начала и в уравнение узла-конца стержня. Соответствующие направляющие косинусы имеют противоположные знаки. Разрешающая матрица размером $M \times M$ системы содержит направляющие косинусы стержней.

Вводится вектор свободных членов системы V . В него заносится информация о нагрузках. Горизонтальные и вертикальные нагрузки образуют отдельные списки P_x и P_y . Даже тогда, когда нагрузка единственная, как в данном случае, ее значение оформляется в виде двух списков длиной 1. Длина списка (число векторов сил) вычисляется оператором `ops`. Списки P_x и P_y должны иметь одинаковую длину.

Решение системы получается с помощью оператора `LinearSolve` пакета `LinearAlgebra`. Но этот способ решения системы линейных уравнений в Maple не единственный. Можно, например, найти обратную матрицу G^{-1} , а затем умножить ее на вектор свободных членов: $S := -G^{-1} \cdot V$. Здесь точка — знак умножения матрицы на матрицу или матрицы на вектор. При этом не требуется загружать пакет `LinearSolve`. Обратная матрица в Maple может быть также найдена с помощью простого деления: $1/G$.

Программу оформим в виде маплета. Общая структура программы следующая:

```
FermaModule:=module()
  Frm:=proc()
    end proc;

  runFerma:=proc()
```

¹ Алгоритм Горячева В.Н., Резунова А.В. (ВГАСУ, Воронеж).