

فرض کنید یک انبار داریم که باید برای 3 هفته آینده خود بر اساس میزان تقاضای مشتری برنامه ریزی کند. مقدار تقاضا برای سه هفته ی آینده مشخص و برابر با جدول زیر می باشد:

هفته اول	هفته دوم	هفته سوم
5 واحد	10 واحد	5 واحد

انبار می تواند در هر هفته حداقل 0 و حداکثر 10 واحد کالا را سفارش دهد و به ازای هر بار سفارشی که انبار انجام میدهد باید مقدار 100 واحد پولی را بپردازد (فارغ از مقدار سفارش) یعنی اگر فقط یک واحد سفارش دهد باید 100 واحد پولی بپردازد و اگر 10 واحد هم سفارش دهد باید 100 واحد پولی بپردازد. فقط در صورتی که مقدار سفارش صفر باشد هیچ هزینه ای نداریم.

هر مقدار از کالا که در هر هفته در انبار باقی بماند به اندازه 5 واحد پولی ضربه در مقدار باقی مانده در انبار در آن هفته باید هزینه بپردازیم (هزینه نگه داری)

میخواهیم تعداد سفارش کالا در هر هفته را پیدا کنیم به طوری که کل هزینه ها حداقل گردد.

توجه داشته باشیم که مقدار سفارش میتواند 0 یا 1 یا 2 یا 3 الی 10 باشد.

این مساله باید به صورت یک مدل برنامه ریزی پویا یا همان توابع بازگشتی (recursive) حل گردد.

یک فرمول یا بیان ریاضی از مساله به صورت زیر است:

$t =$  تعداد هفته ها (در اینجا برابر 3)

$d(t) = t$  مقدار تقاضا در هفته

$i(t) = t$  موجودی انبار در ابتدای هفته

$x(t) = t$  مقدار سفارش در هفته

$O(x) = \begin{cases} 0, & x < 0 \\ 100, & x \geq 0 \end{cases}$  تابع هزینه ی سفارش

$f_t(i_t)$ : تابع هزینه کل برای هفته  $t$  و موجودی  $i_t$

برای مثال  $f_3(0)$  یعنی مقدار هزینه برای هفته سوم با موجودی صفر.

چون مساله باید به صورت بازگشتی حل گردد به تعریف یک هفته فرضی چهارم با هزینه صفر نیاز داریم:

$f_4(*) = 0$

(برای مثال اگر دنباله ی فیبوناچی را بخواهیم به صورت تابع بازگشتی حل کنیم  $f_0 = 0, f_1 = 1$  باید باشد اینجا هم این  $f_4$  همین نقش را بازی میکند)

مقدار تابع کل هزینه به صورت ریاضی به شرح زیر است:

$f_t(i_t) = \operatorname{argmin}(x(t)) = \{O(X) + \operatorname{MAX}(5 * (i_t + x_t - d_t), 0) + f_{t+1}(i_t + x_t - d_t)\}$

$\operatorname{Argmin}(x(t))$  یعنی  $X(t)$  را پیدا کن به طوری که تابع مقابل را حداقل کند.