Algorithm 1: Standard preconditioned conjugate gradients (PCG).

Input : `mvm_A()` – function for matrix-vector multiplication (MVM) with matrix A
$\mathbf{b}$ – vector to solve against
$P^{-1}()$ – function for preconditioner

Output : $A^{-1}\mathbf{b}$.

$\mathbf{u}_0 \leftarrow \mathbf{0}$ // Current solution
$\mathbf{r}_0 \leftarrow$ `mvm_A(u0) - b` // Current error $\quad r_0 \leftarrow b - \text{mvm\_}A(u_0)$
$\mathbf{z}_0 \leftarrow P^{-1}(\mathbf{r}_0)$ // Preconditioned error
$\mathbf{d}_0 \leftarrow \mathbf{z}_0$ // "Search" direction for next solution

for $j \leftarrow 0$ **to** T **do**
  $\mathbf{v}_j \leftarrow$ `mvm_A(` $\mathbf{d}_{j-1}$ `)`
  $\alpha_j \leftarrow (\mathbf{r}_{j-1}^\top \mathbf{z}_{j-1})/(\mathbf{d}_{j-1}^\top \mathbf{v}_j)$
  $\mathbf{u}_j \leftarrow \mathbf{u}_{j-1} + \alpha_j \mathbf{d}_{j-1}$
  $\mathbf{r}_j \leftarrow \mathbf{r}_{j-1} - \alpha_j \mathbf{v}_j$
  if $\|\mathbf{r}_j\|_2 <$ tolerance **then return** $\mathbf{u}_j$;
  $\mathbf{z}_j \leftarrow P^{-1}(\mathbf{r}_j)$
  $\beta_j \leftarrow (\mathbf{z}_j^\top \mathbf{z}_j)/(\mathbf{z}_{j-1}^\top \mathbf{z}_{j-1})$ $\quad \beta_j \leftarrow (z_j^T r_j)/(z_{j-1}^T r_{j-1})$
  $\mathbf{d}_j \leftarrow \mathbf{z}_j - \beta_j \mathbf{d}_{j-1}$ $\quad d_j \leftarrow (z_j + \beta_j d_{j-1})$
end

return $\mathbf{u}_{j+1}$