

一种程序执行的方法、装置、存储介质及电子设备

技术领域

[01]本申请涉及计算机领域，特别涉及一种程序执行的方法、装置、存储介质及电子设备。

背景技术

[02]随着互联网的发展，隐私性与安全性逐渐成为互联网产品的核心要素。

[03]当执行某些程序时，程序中的相关数据可能会因为程序的执行而被泄露，例如，执行循环操作时，与该操作对应的程序需要重复运行十次，该运行数据虽然可以进行加密使操作者无法直接知悉该程序具体需要运行几次，但是，操作者可以通过观察程序运行的次数推测出该运行数据的大概范围，那么，程序运行的次数就会被泄露，降低了数据的隐私性。

[04]所以，如何在程序执行时保证数据的安全性是一个亟待解决的问题。

发明内容

[05]本说明书提供一种程序执行的方法、装置、存储介质及电子设备，以至少部分的解决上述问题。

[06]本说明书采用下述技术方案：本说明书提供了一种程序执行的方法，所述方法包括：确定多个待执行程序中的关键数值；针对每个关键数值，为该关键数值设置对应于该关键数值的第一空槽；比较该关键数值与其他关键数值的大小关系；根据比较结果，重新确定各第一空槽与该关键数值的对应关系；根据重新确定的该关键数值对应的第一空槽，对该关键数值进行分裂，得到该关键数值对应的各分裂数值，其中，各分裂数值之和为该关键数值；将该关键数值所在的待执行程序作为原始程序，并针对每个分裂数值，采用该分裂数值替代所述原始程序中的该关键数值，得到该原始程序对应的分裂程序；执行所有待执行程序对应的分裂程序。

[07]可选地，为该关键数值设置对应于该关键数值的第一空槽，具体包括：为该关键数值设置对应于该关键数值的第一空槽以及用于存储该关键数值的第二空槽；根据重新确定的该关键数值对应的第一空槽，对该关键数值进行分裂，具体包括：根据重新确定的该关键数值对应的第一空槽以及用于存储该关键数值的第二空槽，对该关键数值进行分裂。

[08]可选地，比较该关键数值与其他关键数值的大小关系，具体包括：将各关键数值作为二叉树的叶子节点；将各叶子节点作为待比较节点；针对具有相同父节点的两个待比较节点，比较该两个待比较节点的大小关系，并根据比较结果确定该两个待比较节点的父节点对应的关键数值；将所述父节点重新作为待比较节点，直至确定出根节点对应的关键数值为止。

[09]可选地，根据比较结果确定该两个待比较节点的父节点对应的关键数值，具体包括：将该两个待比较节点对应的关键数值中的较大者作为该两个待比较节点的父节点对应的关键数值。

[10]可选地，重新确定各第一空槽与该关键数值的对应关系，具体包括：针对所述二叉树的每个父节点，确定该父节点的子节点在所述二叉树中所在的层数，并根据所述层数确定该父节点的子节点对应的阈值；若该父节点的两个子节点对应的关键数值中的较大者与较小者的比值不小于所述阈值，则确定该父节点在所述二叉树中对应的子树中包含的所有叶子节点对应的关键数值，作为指定关键数值，并将各指定关键数值对应的第一空槽重新设置为均对应于该父节点对应的关键数值。

[11]可选地，根据重新确定的该关键数值对应的第一空槽以及用于存储该关键数值的第二空槽，对该关键数值进行分裂，具体包括：将该关键数值对应的第一空槽以及用于存储该关键数值的第二空槽均作为该关键数值对应的可用空槽；根据该关键数值对应的可用空槽的数量，将该关键数值分裂成所述数量的分裂数值，使从该关键数值分裂出的各分裂数值之间的差值最小，其中，每个可用空槽用于存储一个分裂数值。

[12]可选地，执行所有待执行程序对应的分裂程序之前，所述还包括：将各分裂程序重新确定为待执行程序，并继续对重新确定出的各待执行程序中的关键数值进行分裂，得到相应的分裂程序，直至满足指定条件为止。

[13]可选地，满足指定条件，具体包括：每个分裂程序中包含的关键数值之间的差值最小；或者，重新确定待执行程序的次数达到指定次数，其中，所述指定次数是根据初始确定的各待执行程序中的关键数值的最大值和平均值确定的。

[14]可选地，确定多个待执行程序中的关键数值，具体包括：确定多个待执行程序中的加密后的关键数值；比较该关键数值与其他关键数值的大小关系，具体包括：采用密态操作的方式，比较该关键数值与其他关键数值的大小关系；重新确定各第一空槽与该关键数值的对应关系，具体包括：采用密态操作的方式，重新确定各第一空槽与该关键数

值的对应关系；对该关键数值进行分裂，具体包括：采用密态操作的方式，对该关键数值进行分裂。

[15]本说明书提供了一种程序执行的装置，所述装置包括：确定模块，用于确定多个待执行程序中的关键数值；处理模块，用于针对每个关键数值，为该关键数值设置对应于该关键数值的第一空槽；比较模块，用于比较该关键数值与其他关键数值的大小关系；对应模块，用于根据比较结果，重新确定各第一空槽与该关键数值的对应关系；分裂模块，用于根据重新确定的该关键数值对应的第一空槽，对该关键数值进行分裂，得到该关键数值对应的各分裂数值，其中，各分裂数值之和为该关键数值；替换模块，用于将该关键数值所在的待执行程序作为原始程序，并针对每个分裂数值，采用该分裂数值替代所述原始程序中的该关键数值，得到该原始程序对应的分裂程序；执行模块，用于执行所有待执行程序对应的分裂程序。

[16]本说明书提供了一种计算机可读存储介质，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述程序执行的方法。

[17]本说明书提供了一种电子设备，包括储存器、处理器及存储在储存器上并可在处理器上运行的计算机程序，所述处理器执行所述程序时实现上述程序执行的方法。

[18]本说明书采用的上述至少一个技术方案能够达到以下有益效果：在本说明书提供的程序执行的方法中，为从待执行程序中确定的每个关键数值设置相应的第一空槽，并比较各关键数值之间的大小关系，根据比较结果重新确定每个关键数值与第一空槽的对应关系，通过第一空槽与用于存储每个关键数值的第二空槽的数量总和，分裂关键数值从而得到各分裂数值，将待执行程序中的关键数值替换为分裂数值，得到分裂程序。

[19]从上述方法可以看出，由待执行程序得到多个分裂程序，因此，当多个待执行程序同时运行时，操作者即使可以推测程序运行的总次数，也无法知悉单一程序运行的具体次数，从而保护了数据的隐私性。

附图说明

[20]此处所说明的附图用来提供对本说明书的进一步理解，构成本说明书的一部分，本说明书的示意性实施例及其说明用于解释本说明书，并不构成对本说明书的不当限定。在附图中：

[21]图 1 为本说明书中一种程序执行方法的流程示意图；

[22]图 2 为本说明书中提供的一种各关键数值比较大小关系的示意图；

[23]图 3 为本说明书中提供的一种关键数值分裂为各分裂数值的示意图；

[24]图 4 为本说明书提供的一种程序执行装置的示意图；

[25]图 5 为本说明书提供的对应于图 1 的电子设备示意图。

具体实施方式

[26]本方案的核心发明思路是：为多个待执行程序中的各关键数值设置对应的空槽，所述各关键数值通过比较后争抢空槽用于分裂，关键数值越大，则其争抢的空槽就越多，所述关键数值分裂后的数量就会越多，关键数值分裂后得到一个或多个分裂数值，将待执行程序中的关键数值替换为分裂数值，得到分裂程序，则可以由一个待执行程序通过关键数值的分裂得到多个分裂程序，当多个待执行程序同时运行时，即使操作者能够推测出所述程序运行的总次数，也无法知悉单个程序具体的运行次数，从而达到了保护数据安全的目的。

[27]为使本说明书的目的、技术方案和优点更加清楚，下面将结合本说明书具体实施例及相应的附图对本说明书技术方案进行清楚、完整地描述。显然，所描述的实施例仅是本说明书一部分实施例，而不是全部的实施例。基于本说明书中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本申请保护的范围。

[28]以下结合附图，详细说明本说明书各实施例提供的技术方案。

[29]图 1 为本说明书提供的一种程序执行方法的流程示意图，具体包括以下步骤 S100 至 S106。

[30]S100：确定多个待执行程序中的关键数值。

[31]本说明书提供的程序执行的执行主体可以是服务器，也可以是诸如个人计算机（Personal Computer，PC）、手机等电子设备，为了便于描述，下面仅以服务器为执行主体，对本说明书提供的程序执行的方法进行说明。

[32]在本说明书实施例中，关键数值是指具有隐私性但在程序运行时会被泄露的数值，比如，待执行程序用于复制操作，复制的次数不希望被操作者知悉，为完成该操作所述待执行程序需要运行 10 次，虽然在待执行程序中循环执行的次数“10”可以是加密的，但是，一旦服务器开始执行该待执行程序，操作者可以通过观察所述待执行程序的运行推测出复制的次数，其中，10 就是所述待执行程序的关键数值。

[33]由于本说明书中的程序执行方法应用于多个待执行程序同时执行的场景，且每个待执行程序中关键数值的数量不定，可以是一个，也可以是多个，因此，以下内容以多个关键数值中的一个关键数值为例进行说明。

[34]S101：针对每个关键数值，为该关键数值设置对应于该关键数值的第一空槽。

[35]当从待执行程序中确定该关键数值后，为该关键数值设置对应的第一空槽，所述第一空槽用于存储该关键数值分裂的分裂数值，其中，第一空槽是指存储空间，每个第一空槽只能存储一个分裂数值，设置第一空槽后，比较该关键数值与其他关键数值的大小关系，并根据比较结果重新分配该关键数值对应的第一空槽，该关键数值越大，分配给该关键数值的第一空槽数量越多。

[36]S102：比较该关键数值与其他关键数值的大小关系。

[37]为该关键数值分配第一空槽后，比较该关键数值与其他关键数值的大小关系，具体的，可通过二叉树对各关键数值进行比较，将各关键数值作为叶子节点，将各叶子节点作为待比较节点，针对具有相同父节点的两个待比较节点，比较该两个待比较节点的大小关系，将两个待比较节点对应的关键数值中的较大者作为所述两个待比较节点的父节点对应的关键数值，将所述父节点重新作为待比较节点，直至确定出根节点对应的关键数值为止，如图2所示。

[38]在图2中，c1、c2、c3、c4分别为四个待执行程序中的关键数值，首先为c1、c2、c3、c4设置对应的第一空槽，然后开始比较c1、c2、c3、c4的大小关系，将这四个关键数值作为叶子节点（即图2中第0层的4个叶子节点），每个叶子节点作为待比较节点，具有相同父节点的两个叶子节点进行比较，如c1和c2比较、c3和c4比较，将c1和c2中的较大者作为二者的父节点对应的关键数值，如c2为3，c1为2，c2大于c1，则c1与c2的父节点对应的关键数值为c2=3，c3与c4同理，将所述父节点（即图2中第1层的2个节点）重新作为待比较节点，直至确定出根节点对应的关键数值为止，如将c2和c4再次进行比较，将较大者c4作为c2和c4的父节点，也即，c4为该二叉树的根节点（即图2中第2层的根节点）。

[39]S103：根据比较结果，重新确定各第一空槽与该关键数值的对应关系。

[40]通过确定具有相同父节点的子节点所在二叉树的层数，进而确定所述子节点对应的阈值，同一层的子节点具有相同的阈值，每层阈值为 2^i ，其中，i为二叉树的层数。

[41]若所述两个待比较节点的关键数值的比值不小于该两个待比较节点对应的阈值，则

确定所述父节点在所述二叉树中对应的子树中包含的所有叶子节点对应的关键数值，作为指定关键数值，并将各指定关键数值对应的第一空槽重新设置为均对应于所述父节点对应的关键数值。

[42]若所述两个待比较节点的关键数值的比值小于该两个待比较节点对应的阈值，则保持父节点在所述二叉树中对应的子树中包含的所有叶子节点与第一空槽的对应关系不变。

[43]在图 2 中，阈值与层数有关，同一层的节点阈值相同，因此第一层阈值为 $2^1=2$ ，c1 与 c2 的比值小于该层阈值 2，所以 c1 与 c2 不争抢第一空槽，保持原有的对应关系不变；c3 与 c4 的比值大于该层阈值 2，则 c4 要争抢 c3 的第一空槽，以使其获取更多的分裂机会，同理，两个父节点 c2 与 c4 继续比较，此时阈值随层数发生变化，第二层阈值为 $2^2=4$ ，c4 与 c2 的比值大于该层阈值 4，则 c4 争抢 c2 所在节点在二叉树中对应的子树中包含的所有叶子节点的第一空槽，最终，c1、c2、c3 对应的第一空槽均提供给 c4 分裂使用，c4 对应于原本为 c1、c2、c3、c4 设置的第一空槽。

[44]S104：根据重新确定的该关键数值对应的第一空槽，对该关键数值进行分裂，得到该关键数值对应的各分裂数值，其中，各分裂数值之和为该关键数值。

[45]在 S101 中设置第一空槽的同时设置用于存储该关键数值的第二空槽，第二空槽也是存储空间，一个第二空槽仅存储一个关键数值，根据重新确定的各第一空槽与该关键数值的对应关系，确定该关键数值的可用空槽的数量，其中，可用空槽为第二空槽与重新分配的第一空槽的数量之和，根据该关键数值对应的可用空槽的数量，将该关键数值分裂成所述数量的分裂数值，使从该关键数值分裂出的各分裂数值之间的差值最小，即，使得到的各分裂数值尽量均匀，其中，每个可用空槽用于存储一个分裂数值，可用空槽有几个，该关键数值就分裂成几个分裂数值，分裂数值之和等于该关键数值。

[46]在图 3 中，根据重新确定的第一空槽与关键数值的对应关系，可知 c1、c2、c3 将其对应的第一空槽均提供给 c4 分裂使用，c1、c2、c3 的第二空槽分别用于存放 c1、c2、c3，所以不提供给 c4 使用，因此，c4 可以用于分裂的空槽包括原本对应于 c1、c2、c3 的第一空槽和对应于 c4 的第一空槽以及存储 c4 的第二空槽，可用空槽的个数为 5 个，所以将 $c4=21$ 尽量均匀的分成五个分裂数值，得到 $D1=4$ 、 $D2=4$ 、 $D3=4$ 、 $D4=4$ 、 $D5=5$ 。

[47]S105：将该关键数值所在的待执行程序作为原始程序，并针对每个分裂数值，采用该分裂数值替代所述原始程序中的该关键数值，得到该原始程序对应的分裂程序。

[48]通过将原始程序中的关键数值替换为由该关键数值分裂出的分裂数值，可以由一个原始程序确定多个分裂程序，使得多个待执行程序在运行时，即使可以推测出程序运行总数，也无法知悉单个程序具体运行的次数，有效保护了数据的安全。

[49]在图 3 中，将 c4 所在的待执行程序作为原始程序，采用 D1、D2、D3、D4、D5 分别替换原始程序中的 c4 的方式，得到相应的分裂程序，通过 5 个分裂数值替换关键数值得到五个分裂程序，原本四个待执行程序变为三个待执行程序 and 五个分裂程序，程序总数量由 4 个变为 8 个。

[50]S106: 执行所有待执行程序对应的分裂程序。

[51]在本说明书提供的程序执行的方法中，为从待执行程序中确定的每个关键数值设置相应的第一空槽，并比较各关键数值之间的大小关系，根据比较结果重新确定每个关键数值与第一空槽的对应关系，关键数值越大，重新确定的与其对应的第一空槽的数量越多，通过第一空槽与用于存储每个关键数值的第二空槽的数量之和，分裂关键数值从而得到各分裂数值，将待执行程序中的关键数值替换为分裂数值，得到分裂程序，则可以由一个待执行程序通过关键数值的分裂得到多个分裂程序，因此，当多个待执行程序同时运行时，操作者即使可以推测程序运行的总次数，也无法知悉单一程序运行的具体次数，解决了在程序运行过程中会泄露具有隐私性数值的问题，保护了数据的安全。

[52]在本说明书实施例中，采用上述方法将关键数值分裂成差值最小的各分裂数值而不是直接将关键数值均分，原因在于操作方可以观测到关键数值平均分裂成多个分裂数值的动作，就可以知道该关键数值被均分成几个分裂数值，即，大概可以推测出关键数值的范围，所以均分关键数值的方法无法保证数据的安全。为了保证数据的安全，虽然也可以采用排序的方式对关键数值进行分裂，即，按从大到小的顺序将各关键数值进行排序，并对排在前 n 位的关键数值进行分裂，然后再次对分裂后的分裂数值以及未经过分裂的关键数值进行排序，并继续分裂，但是这种方式每分裂一次，就需要进行一次全量的排序，而排序这种操作在计算机内部时比较消耗计算资源且效率较低的，因此，如本说明书中采用二叉树的方式对关键数值进行比较并将关键数值分裂成差值最小的各分裂数值，既可以提高效率又能够保证数据的安全。

[53]在图 3 中，分别为从四个待执行程序中确定的关键数值 c1、c2、c3、c4 设置相应的第一空槽，比较 c1、c2、c3、c4 的大小，根据比较的结果可确定 c1、c2、c3 将各自的第一空槽提供给 c4 使用，c4 最终有 5 个可用空槽，分裂成尽量均匀的 5 个分裂数值 D1、D2、D3、D4、D5，将分裂数值替换关键数值得到五个相对应的分裂程序，上述操作均

在加密状态下完成，由四个待执行程序变为八个待执行程序，使操作者无法知悉哪个关键数值进行了分裂以及分裂的关键数值的分裂次数，因此，当多个待执行程序同时运行时，操作者即使可以知悉程序运行的总次数，也无法知悉单一程序运行的具体次数，保护了数据的安全。

[54]在本说明书实施例中，各关键数值的确定是在加密状态下进行，即，操作方对关键数值不可见，且采用密态操作的方式，比较关键数值的大小关系，并根据比较结果重新确定第一空槽与关键数值的对应关系，同时，关键数值的分裂也采用密态操作的方式，若上述操作在非加密状态下进行，操作者则可以通过观察多个待执行程序中的一个待执行程序运行的次数以及分裂的次数，推测出关键数值的大小，因此，需要采取加密的方式完成上述操作才能保证数据的安全性。

[55]本说明书实施例中的多个待执行程序是否分裂由各关键数值的比较结果决定，多个待执行程序可能分裂也可能不分裂，因此，为进一步保障数据的安全，需使分裂后得到的各分裂程序与未分裂的待执行程序的关键数值尽量均匀，则如图 1 所示的流程，在步骤 S106 之前，根据各分裂数值获得各分裂程序后，将各分裂程序重新确定为步骤 S100 中的待执行程序，直至各分裂程序中包含的关键数值尽量均匀，即，各分裂程序中包含的关键数值之间差值最小或者重新确定待执行程序的次数达到指定次数时，就可以停止分裂，其中，指定次数是指初始确定的各待执行程序中的关键数值的最大值和平均值的比值或预先设定的次数。

[56]经过将关键数值大的待执行程序进行分裂得到关键数值均匀的各分裂程序与未分裂的待执行程序再次作为待执行程序进行关键数值的比较并分裂，使得最终确定的各程序的关键数值均匀，进一步保证了程序运行过程中数据的安全性。

[57]以上为本说明书的一个或多个实施例提供的程序执行方法，基于同样的思路，本说明书还提供了相应的程序执行装置，如图 4 所示。

[58]图 4 为本说明书提供的一种程序执行装置示意图，具体包括：确定模块 401，用于确定多个待执行程序中的关键数值；处理模块 402，用于针对每个关键数值，为该关键数值设置对应于该关键数值的第一空槽；比较模块 403，用于比较该关键数值与其他关键数值的大小关系；对应模块 404，用于根据比较结果，重新确定各第一空槽与该关键数值的对应关系；分裂模块 405，用于根据重新确定的该关键数值对应的第一空槽，对该关键数值进行分裂，得到该关键数值对应的各分裂数值，其中，各分裂数值之和为该关键数值；替换模块 406，用于将该关键数值所在的待执行程序作为原始程序，并针对

每个分裂数值，采用该分裂数值替代所述原始程序中的该关键数值，得到该原始程序对应的分裂程序；执行模块 407，用于执行所有待执行程序对应的分裂程序。

[59]可选地，所述处理模块 402 具体用于，为该关键数值设置对应于该关键数值的第一空槽以及用于存储该关键数值的第二空槽；所述分裂模块 405 具体用于，根据重新确定的该关键数值对应的第一空槽以及用于存储该关键数值的第二空槽，对该关键数值进行分裂。

[60]可选地，所述比较模块 403 具体用于，将各关键数值作为二叉树的叶子节点；将各叶子节点作为待比较节点；针对具有相同父节点的两个待比较节点，比较该两个待比较节点的大小关系，并根据比较结果确定该两个待比较节点的父节点对应的关键数值；将所述父节点重新作为待比较节点，直至确定出根节点对应的关键数值为止。

[61]可选地，所述比较模块 403 具体用于，将该两个待比较节点对应的关键数值中的较大者作为该两个待比较节点的父节点对应的关键数值。

[62]可选地，所述对应模块 404 具体用于，针对所述二叉树的每个父节点，确定该父节点的子节点在所述二叉树中所在的层数，并根据所述层数确定该父节点的子节点对应的阈值；若该父节点的两个子节点对应的关键数值中的较大者与较小者的比值不小于所述阈值，则确定该父节点在所述二叉树中对应的子树中包含的所有叶子节点对应的关键数值，作为指定关键数值，并将各指定关键数值对应的第一空槽重新设置为均对应于该父节点对应的关键数值。

[63]可选地，所述分裂模块 405 具体用于，将该关键数值对应的第一空槽以及用于存储该关键数值的第二空槽均作为该关键数值对应的可用空槽；根据该关键数值对应的可用空槽的数量，将该关键数值分裂成所述数量的分裂数值，使从该关键数值分裂出的各分裂数值之间的差值最小，其中，每个可用空槽用于存储一个分裂数值。

[64]可选地，所述分裂模块 405 具体用于，在执行所有待执行程序对应的分裂程序之前，将各分裂程序重新确定为待执行程序，并继续对重新确定出的各待执行程序中的关键数值进行分裂，得到相应的分裂程序，直至满足指定条件为止。

[65]可选地，满足指定条件，具体包括：每个分裂程序中包含的关键数值之间的差值最小；或者，重新确定待执行程序的次数达到指定次数，其中，所述指定次数是根据初始确定的各待执行程序中的关键数值的最大值和平均值确定的。

[66]可选地，所述确定模块 401 具体用于，确定多个待执行程序中的加密后的关键数值；

所述比较模块 403 具体用于,采用密态操作的方式,比较该关键数值与其他关键数值的大小关系;所述对应模块 404 具体用于,采用密态操作的方式,重新确定各第一空槽与该关键数值的对应关系;所述分裂模块 405 具体用于,采用密态操作的方式,对该关键数值进行分裂。

[67]本说明书还提供了一种计算机可读存储介质,该存储介质存储有计算机程序,计算机程序可用于执行上述图 1 提供的程序执行方法。

[68]本说明书还提供了图 5 所示的电子设备的结构示意图。如图 5 所述,在硬件层面,该无人驾驶设备包括处理器、内部总线、网络接口、内存以及非易失性存储器,当然还可能包括其他业务所需要的硬件。处理器从非易失性存储器中读取对应的计算机程序到内存中然后运行,以实现上述图 1 所述的判断事件执行顺序方法。当然,除了软件实现方式之外,本说明书并不排除其他实现方式,比如逻辑器件抑或软硬件结合的方式等等,也就是说以下处理流程的执行主体并不限于各个逻辑单元,也可以是硬件或逻辑器件。

[69]在 20 世纪 90 年代,对于一个技术的改进可以很明显地区分是硬件上的改进(例如,对二极管、晶体管、开关等电路结构的改进)还是软件上的改进(对于方法流程的改进)。然而,随着技术的发展,当今的很多方法流程的改进已经可以视为硬件电路结构的直接改进。设计人员几乎都通过将改进的方法流程编程到硬件电路中来得到相应的硬件电路结构。因此,不能说一个方法流程的改进就不能用硬件实体模块来实现。例如,可编程逻辑器件(Programmable Logic Device, PLD)(例如现场可编程门阵列(Field Programmable Gate Array, FPGA))就是这样一种集成电路,其逻辑功能由用户对器件编程来确定。由设计人员自行编程来把一个数字系统“集成”在一片 PLD 上,而不需要请芯片制造厂商来设计和制作专用的集成电路芯片。而且,如今,取代手工地制作集成电路芯片,这种编程也多半改用“逻辑编译器(logic compiler)”软件来实现,它与程序开发撰写时所用的软件编译器相类似,而要编译之前的原始代码也得用特定的编程语言来撰写,此称之为硬件描述语言(Hardware Description Language, HDL),而 HDL 也并非仅有一种,而是有许多种,如 ABEL(Advanced Boolean Expression Language)、AHDL(Altera Hardware Description Language)、Confluence、CUPL(Cornell University Programming Language)、HDCal、JHDL(Java Hardware Description Language)、Lava、Lola、MyHDL、PALASM、RHDL(Ruby Hardware Description Language)等,目前最普遍使用的是 VHDL(Very-High-Speed Integrated Circuit Hardware Description Language)与 Verilog。本领域技术人员也应该清楚,只需要将方法流程用上述几种硬件描述语言

稍作逻辑编程并编程到集成电路中，就可以很容易得到实现该逻辑方法流程的硬件电路。

[70]控制器可以按任何适当的方式实现，例如，控制器可以采取例如微处理器或处理器以及存储可由该（微）处理器执行的计算机可读程序代码（例如软件或固件）的计算机可读介质、逻辑门、开关、专用集成电路（Application Specific Integrated Circuit, ASIC）、可编程逻辑控制器和嵌入微控制器的形式，控制器的例子包括但不限于以下微控制器：ARC 625D、Atmel AT91SAM、Microchip PIC18F26K20 以及 Silicone Labs C8051F320，存储器控制器还可以被实现为存储器的控制逻辑的一部分。本领域技术人员也知道，除了以纯计算机可读程序代码方式实现控制器以外，完全可以通过将方法步骤进行逻辑编程来使得控制器以逻辑门、开关、专用集成电路、可编程逻辑控制器和嵌入微控制器等的形式来实现相同功能。因此这种控制器可以被认为是一种硬件部件，而对其内包括的用于实现各种功能的装置也可以视为硬件部件内的结构。或者甚至，可以将用于实现各种功能的装置视为既可以是实现方法的软件模块又可以是硬件部件内的结构。

[71]上述实施例阐明的系统、装置、模块或单元，具体可以由计算机芯片或实体实现，或者由具有某种功能的产品来实现。一种典型的实现设备为计算机。具体的，计算机例如可以为个人计算机、膝上型计算机、蜂窝电话、相机电话、智能电话、个人数字助理、媒体播放器、导航设备、电子邮件设备、游戏控制台、平板计算机、可穿戴设备或者这些设备中的任何设备的组合。

[72]为了描述的方便，描述以上装置时以功能分为各种单元分别描述。当然，在实施本说明书时可以把各单元的功能在同一个或多个软件和/或硬件中实现。

[73]本领域内的技术人员应明白，本申请的实施例可提供为方法、系统、或计算机程序产品。因此，本申请可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且，本申请可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质（包括但不限于磁盘存储器、CD-ROM、光学存储器等）上实施的计算机程序产品的形式。

[74]本申请是参照根据本申请实施例的方法、设备（系统）、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器，使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中

指定的功能的装置。

[75]这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中，使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品，该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[76]这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上，使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理，从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[77]在一个典型的配置中，计算设备包括一个或多个处理器(CPU)、输入/输出接口、网络接口和内存。

[78]内存可能包括计算机可读介质中的非永久性存储器，随机存取存储器(RAM)和/或非易失性内存等形式，如只读存储器(ROM)或闪存(flash RAM)。内存是计算机可读介质的示例。

[79]计算机可读介质包括永久性和非永久性、可移动和非可移动媒体可以由任何方法或技术来实现信息存储。信息可以是计算机可读指令、数据结构、程序的模块或其他数据。计算机的存储介质的例子包括，但不限于相变内存(PRAM)、静态随机存取存储器(SRAM)、动态随机存取存储器(DRAM)、其他类型的随机存取存储器(RAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、快闪记忆体或其他内存技术、只读光盘只读存储器(CD-ROM)、数字多功能光盘(DVD)或其他光学存储、磁盒式磁带，磁带磁磁盘存储或其他磁性存储设备或任何其他非传输介质，可用于存储可以被计算设备访问的信息。按照本文中的界定，计算机可读介质不包括暂存电脑可读媒体(transitory media)，如调制的数据信号和载波。

[80]还需要说明的是，术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、商品或者设备不仅包括那些要素，而且还包括没有明确列出的其他要素，或者是还包括为这种过程、方法、商品或者设备所固有的要素。在没有更多限制的情况下，由语句“包括一个……”限定的要素，并不排除在包括所述要素的过程、方法、商品或者设备中还存在另外的相同要素。

[81]本领域技术人员应明白，本说明书的实施例可提供为方法、系统或计算机程序产品。

因此，本说明书可采用完全硬件实施例、完全软件实施例或结合软件和硬件方面的实施例的形式。而且，本说明书可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质（包括但不限于磁盘存储器、CD-ROM、光学存储器等）上实施的计算机程序产品的形式。

[82]本说明书可以在由计算机执行的计算机可执行指令的一般上下文中描述，例如程序模块。一般地，程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本说明书，在这些分布式计算环境中，由通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中，程序模块可以位于包括存储设备在内的本地和远程计算机存储介质中。

[83]本说明书中的各个实施例均采用递进的方式描述，各个实施例之间相同相似的部分互相参见即可，每个实施例重点说明的都是与其他实施例的不同之处。尤其，对于系统实施例而言，由于其基本相似于方法实施例，所以描述的比较简单，相关之处参见方法实施例的部分说明即可。

[84]以上所述仅为本说明书的实施例而已，并不用于限制本说明书。对于本领域技术人员来说，本说明书可以有各种更改和变化。凡在本说明书的精神和原理之内所作的任何修改、等同替换、改进等，均应包含在本申请的权利要求范围之内。

权利要求书

1、一种程序执行的方法，所述方法包括：

确定多个待执行程序中的关键数值；

针对每个关键数值，为该关键数值设置对应于该关键数值的第一空槽；

比较该关键数值与其他关键数值的大小关系；

根据比较结果，重新确定各第一空槽与该关键数值的对应关系；

根据重新确定的该关键数值对应的第一空槽，对该关键数值进行分裂，得到该关键数值对应的各分裂数值，其中，各分裂数值之和为该关键数值；

将该关键数值所在的待执行程序作为原始程序，并针对每个分裂数值，采用该分裂数值替代所述原始程序中的该关键数值，得到该原始程序对应的分裂程序；

执行所有待执行程序对应的分裂程序。

2、如权利要求 1 所述的方法，为该关键数值设置对应于该关键数值的第一空槽，包括：

为该关键数值设置对应于该关键数值的第一空槽以及用于存储该关键数值的第二空槽；

根据重新确定的该关键数值对应的第一空槽，对该关键数值进行分裂，包括：

根据重新确定的该关键数值对应的第一空槽以及用于存储该关键数值的第二空槽，对该关键数值进行分裂。

3、如权利要求 1 所述的方法，比较该关键数值与其他关键数值的大小关系，包括：

将各关键数值作为二叉树的叶子节点；

将各叶子节点作为待比较节点；

针对具有相同父节点的两个待比较节点，比较该两个待比较节点的大小关系，并根据比较结果确定该两个待比较节点的父节点对应的关键数值；

将所述父节点重新作为待比较节点，直至确定出根节点对应的关键数值为止。

4、如权利要求 3 所述的方法，根据比较结果确定该两个待比较节点的父节点对应的关键数值，包括：

将该两个待比较节点对应的关键数值中的较大者作为该两个待比较节点的父节点对应的关键数值。

5、如权利要求 3 所述的方法，重新确定各第一空槽与该关键数值的对应关系，包括：

针对所述二叉树的每个父节点，确定该父节点的子节点在所述二叉树中所在的层数，

并根据所述层数确定该父节点的子节点对应的阈值;

若该父节点的两个子节点对应的关键数值中的较大者与较小者的比值不小于所述阈值,则确定该父节点在所述二叉树中对应的子树中包含的所有叶子节点对应的关键数值,作为指定关键数值,并将各指定关键数值对应的第一空槽重新设置为均对应于该父节点对应的关键数值。

6、如权利要求 2 所述的方法,根据重新确定的该关键数值对应的第一空槽以及用于存储该关键数值的第二空槽,对该关键数值进行分裂,包括:

将该关键数值对应的第一空槽以及用于存储该关键数值的第二空槽均作为该关键数值对应的可用空槽;

根据该关键数值对应的可用空槽的数量,将该关键数值分裂成所述数量的分裂数值,使从该关键数值分裂出的各分裂数值之间的差值最小,其中,每个可用空槽用于存储一个分裂数值。

7、如权利要求 1 所述的方法,执行所有待执行程序对应的分裂程序之前,所述方法还包括:

将各分裂程序重新确定为待执行程序,并继续对重新确定出的各待执行程序中的关键数值进行分裂,得到相应的分裂程序,直至满足指定条件为止。

8、如权利要求 7 所述的方法,满足指定条件,包括:

每个分裂程序中包含的关键数值之间的差值最小;或者,

重新确定待执行程序的次数达到指定次数,其中,所述指定次数是根据初始确定的各待执行程序中的关键数值的最大值和平均值确定的。

9、如权利要求 1 所述的方法,确定多个待执行程序中的关键数值,包括:

确定多个待执行程序中的加密后的关键数值;

比较该关键数值与其他关键数值的大小关系,包括:

采用密态操作的方式,比较该关键数值与其他关键数值的大小关系;

重新确定各第一空槽与该关键数值的对应关系,包括:

采用密态操作的方式,重新确定各第一空槽与该关键数值的对应关系;

对该关键数值进行分裂,包括:

采用密态操作的方式,对该关键数值进行分裂。

10、一种程序执行的装置,所述装置包括:

确定模块,用于确定多个待执行程序中的关键数值;

处理模块,用于针对每个关键数值,为该关键数值设置对应于该关键数值的第一空

槽;

比较模块, 用于比较该关键数值与其他关键数值的大小关系;

对应模块, 用于根据比较结果, 重新确定各第一空槽与该关键数值的对应关系;

分裂模块, 用于根据重新确定的该关键数值对应的第一空槽, 对该关键数值进行分裂, 得到该关键数值对应的各分裂数值, 其中, 各分裂数值之和为该关键数值;

替换模块, 用于将该关键数值所在的待执行程序作为原始程序, 并针对每个分裂数值, 采用该分裂数值替代所述原始程序中的该关键数值, 得到该原始程序对应的分裂程序;

执行模块, 用于执行所有待执行程序对应的分裂程序。

11、一种计算机可读存储介质, 所述存储介质存储有计算机程序, 所述计算机程序被处理器执行时实现权利要求 1~9 中任一项所述的方法。

12、一种电子设备, 包括储存器、处理器及存储在储存器上并可在处理器上运行的计算机程序, 所述处理器执行所述程序时实现权利要求 1~9 中任一项所述的方法。